European Commission

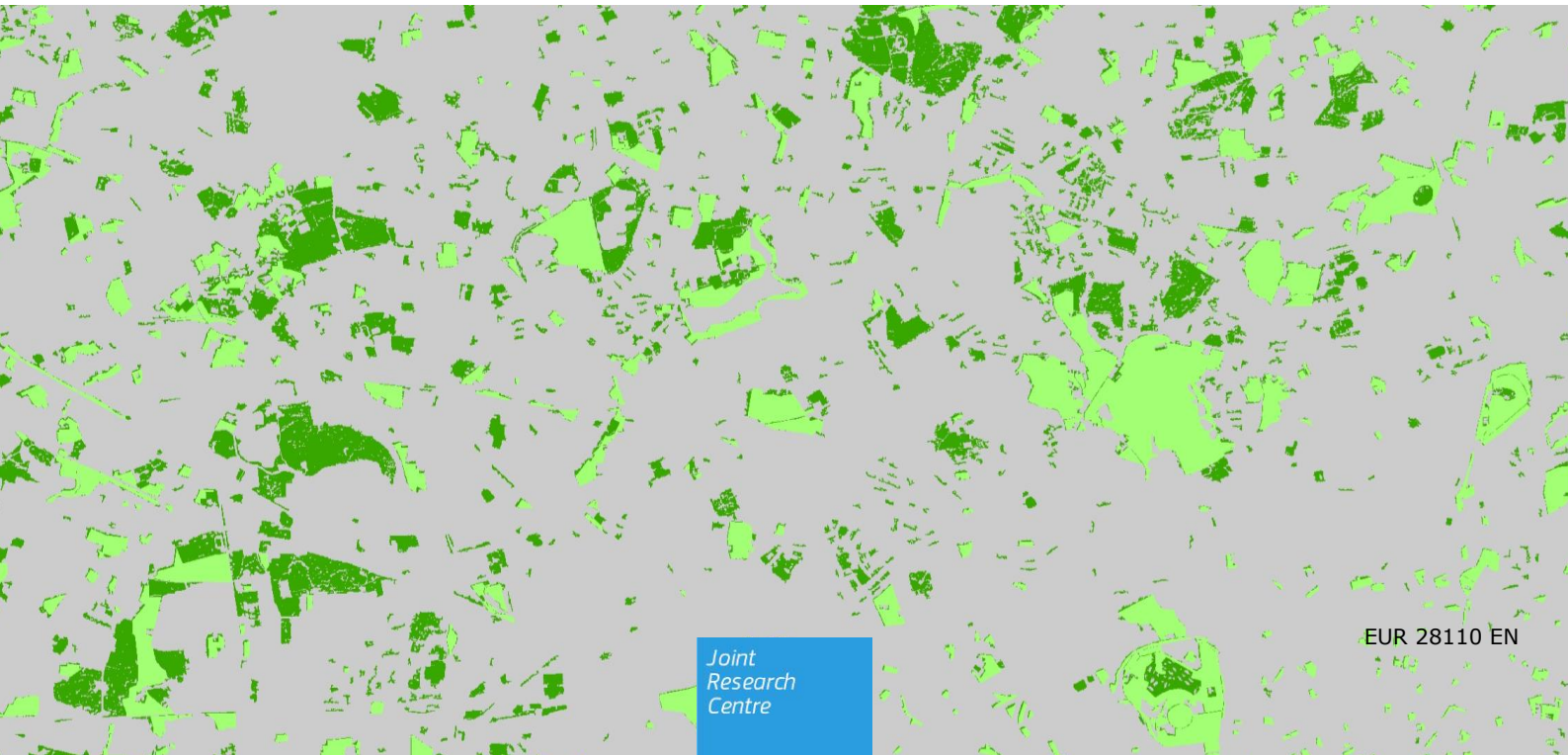
JRC TECHNICAL REPORTS

How green are the European Cities? Exploring the Green European Settlement Map 2016

A systematic comparison between the Green ESM vs. the UA urban green areas in 300 European cities

FERRI Stefano, SIRAGUSA Alice, PAFI Maria, HALKIA Matina

2016



EUR 28110 EN

Joint Research Centre

Contact information
Name: Stefano Ferri
Address: European Commission, DG Joint Research Centre (JRC)
Directorate E Space, Security and Migration
Disaster Risk Management Unit
Via Enrico Fermi 2749, I-21027 Ispra (VA) Italy, TP 267
E-mail: stefano.ferri@jrc.ec.europa.eu
Tel.: +39 0332 786751

JRC Science Hub
https://ec.europa.eu/jrc

JRC102524

EUR 28110 EN

PDF	ISBN 978-92-79-61902-1	ISSN 1831-9424	doi:10.2788/52902
Print	ISBN 978-92-79-61901-4	ISSN 1018-5593	doi:10.2788/24945

Luxembourg: Publications Office of the European Union, 2016

How to cite: FERRI Stefano, SIRAGUSA Alice, PAFI Maria, HALKIA Matina; How green are the European Cities? Exploring the Green European Settlement Map 2016. A systematic comparison between the Green ESM vs. the UA urban green areas in 300 European cities; EUR 28110 EN; doi:10.2788/52902

Table of contents

Acknowledgements

The author would like to acknowledge the contribution of Dr. Alice Siragusa. Her professional experiences as architect and planner provided insight and expertise that greatly assisted the research. Furthermore, thanks to the python code she developed we were able to process 300 cities in a relatively short period.

Abstract

This work aims at contributing in measuring and evaluating urban green areas in European cities. The main objective of the work was to compare the most common and used dataset on cities available in Europe, Urban Atlas, with the new dataset produced by the JRC, the Green European Settlement Map 2016.

The method consists of developing and applying an automatic computation of green urban areas in the 300 cities in which Urban Atlas dataset is available for the year 2006.

This report will illustrate the rationale of the work and results. In particular, it will show that the Green European Settlement Map 2016 detects more green areas than Urban Atlas.

1. Introduction

Urban green areas are receiving growing attention in the international scenario.

The European Commission put in place several initiatives to encourage the promotion and increment of urban green areas: in particular, the European Green Leaf aims at underpinning the work of creating an ever-better environment for European citizens.

The EU also adopted the 7th Environment Action Programme (7th EAP) entitled "Living well, within the limits of our planet". It provides the basis for EU environmental policy up to 2020. This programme aims to enhance Europe's ecological resilience and transform the EU into an inclusive and sustainable green economy[1].

Other international organizations address the same topic as well. Particularly in the new set of Sustainable Development Goals approved by the United Nations in September 2015, a specific target is devoted to urban green spaces [1].

Indeed the SDG 11.7 set the following target:

> "*by 2030, provide universal access to safe, inclusive and accessible, green and public spaces, particularly for women and children, older persons and persons with disabilities".*

The World Health Organization underlines in several documents the contribution of urban green areas in improving the quality of life or urban dwellers [2].

Despite the growing attention devoted to this topic, consistent and comparable data for all European urban areas are not available.

Urban Atlas, which provides information for more than 300 European cities for 2006, does not cover the entire region and does not allow having consistent and comparable data for all the European urban centres.

Recently, the JRC produced a new raster layer called Green European Settlement Map 2016 at 10m resolution, which provides useful information on green areas [3]. The ESM allows policy-makers and researchers to compare information regarding urban green areas across all European countries.

The goal of this study is to compare these two datasets for all cities in which the UA 2006 is available. The report contains a full description of the input data, the methodology and the results. The annexes contain the full Python script, the full table of the results per city and the data aggregated per country.

[1] http://ec.europa.eu/environment/europeangreencapital/europeangreenleaf/index.html

2. Methodology

It is hard to define what urban green areas are. Several different definitions have been used in different datasets and this influence negatively the possibility to compare data among different dataset. For example, in the UA Urban green areas are defined as follows:

*"Feature Class **14100: Green urban areas***

Public green areas for predominantly recreational use such as gardens, zoos, parks, castle parks. Suburban natural areas that have become and are managed as urban parks. Forests or green areas extending from the surroundings into urban areas are mapped as green urban areas when at least two sides are bordered by urban areas and structures, and traces of recreational use are visible.

Not included are: Private gardens within housing areas → class 1.1;

Cemeteries → class 1.2.1;

Buildings within parks, such as castles or museums → class 1.2.1;

Patches of natural vegetation or agricultural areas enclosed by built-up areas without being managed as green urban areas → class 1." [4, p. 21]

Ones of the main issues within this definition regard the minimum mapping unit, which excludes all areas smaller than 0.25 ha, the ownership status of the area (private or public), and its management.

Others, such as Breuste et al, define urban green space *"as a range of parks, street trees, urban agriculture, residential lawns, and roof gardens*" [5]. This definition includes also private green areas and disregards size area.
The definition of green space can also be influenced by local legislation or context.
For example, according to the Romanian legislation (Law no.24/2007), green spaces used for recreation, besides thematic parks like museums and sport facilities, are of three types:
"– The park, defined as a green area of a minimum of one hectare destined for sport and leisure activities;
– The square, defined as a green area smaller than one hectare; and
– The forest, which is an afforested piece of land used for leisure activities" [6, p. 122].
This definition includes both use and minimum size aspect of green area.

Definitions depend on several different elements as local characteristics, regulations, and field of application.
For example, if you consider only social use of the green space, one should consider only public space, or space in public use; if one wants to consider also contribution to the environmental conditions of cities, all green spaces should be included, regardless their size.
Remote sense data, because of their nature, disregards ownership status, size (except because of its resolution), and use. Database such as Green ESM provide a full map of urban green with a high resolution (10m, with input data at 2.5m) and with a wall-to-wall coverage, regardless administrative boundaries.

The following section describes the input data used for the study, the definition of the study area and the description of the workflow[2].

2.1 Input data

Four different datasets have been used to implement the model.

The city boundaries have been defined using the **urban clusters** elaborated by **GEOSTAT in 2006 intersected with the Urban Atlas coverage.** This dataset contains urban clusters, based on local population data of 2006, produced by the Eurostat-GEOSTAT grid. Urban clusters are defined as groups of contiguous raster cells of 1 sqkm size, having a population density of at least 300 inhabitants/sqkm and a total population of at least 5000. The definition of urban clusters underpins the urban/rural typology of NUTS3 regions and the degree of urbanisation classification of local administrative (LAU2) units. Data are available for EU countries (excluding Cyprus and the outermost regions of France), Norway, Switzerland, Leichtenstein and Iceland.[3] For a full explanation of the definition of the study areas, see the following paragraph.

The **European Settlement Map (ESM) 2016 at 10m resolution** has been used to extract the green urban areas. The layers used for this calculation have the following codes: 40 and 41. The layers are two types of green in built-up areas: the layer 40 represents the Green NDVIx in BU Areas, the layer 41 represents those green areas which are not detected by the NDVIx, but are included in the Urban Atlas Green in BU Areas [3, p. 13]; this is due to the definition of urban green area used in different datasets.

The data on green areas obtained using the ESM has been compared with the **European Urban Atlas 2006**, selecting the feature class **14100: Green urban areas** [4]**.** The European Urban Atlas is part of the local component of the GMES/Copernicus land monitoring services. It provides land use maps for 305 Large Urban Zones and their surroundings (more than 100.000 inhabitants as defined by the Urban Audit) for the reference year 2006. The GIS data can be downloaded together with a map for each urban area covered and a report with the metadata. The geometric resolution (Minimum Mapping Unit) is 0.25 ha. The positional accuracy of the dataset is +/- 5 meters. [4]

It has been decided to use the UA 2006, despite the fact that a new version (2012) is available because the 2006 version has been updated and corrected in the last years, while the latest version is still in course of publication.

In order to calculate the total population of the city and the number of inhabitants served with adequate provision of green areas, the **EU 100m pop mosaic**, created by downscaling the Global Human Settlement Layer (GHSL) and the best available input census data [5], have been used.

2.2 Areas of study

The first option to define the study areas was to use the UA, but in most of the cases, it represents a much larger perimeter of what can be considered urban. The type of study

---

[2] The scripting has been completed using the Python language using the ArcPy library for ArcGIS version 10.2.

[3] Data are available here: http://ec.europa.eu/eurostat/web/gisco/geodata/reference-data/population-distribution-demography/clusters#clustershd11

[4] Available for the download for 305 urban centres http://land.copernicus.eu/local/urban-atlas/urban-atlas-2006

that deals with the urban green areas has to be done specifically in the urban context and not in those parts of the city which are inside the administrative boundaries but are not urban.

Then, since the goal of the study is to identify the urban green areas and to compare the information of the ESM with the UA, we have to make sure that the study area is included in the UA boarder and also that is urban. So an intersection of the two dataset has to be done: the UA city border assures that we will have data to compare the ESM; the GEOSTAT urban clusters assure that the study areas have the characteristics of urbanity.

Due to this process, some important cities have been excluded from the study, such as two EU capitals: Nicosia, capital of Cyprus, not covered by Eurostat-GEOSTAT grid, and Zagreb, Croatia, which joined the EU in 2013 and was not covered by UA 2006. At the end of this pre-process 300 cities have been studied.

As example, the differences of the two dataset are showed for the cities of Barcelona, Spain and Malmo, Sweden.

Figure 1 - Difference of city border of UA, Urban cluster and their intersection – City of Barcelona, Spain

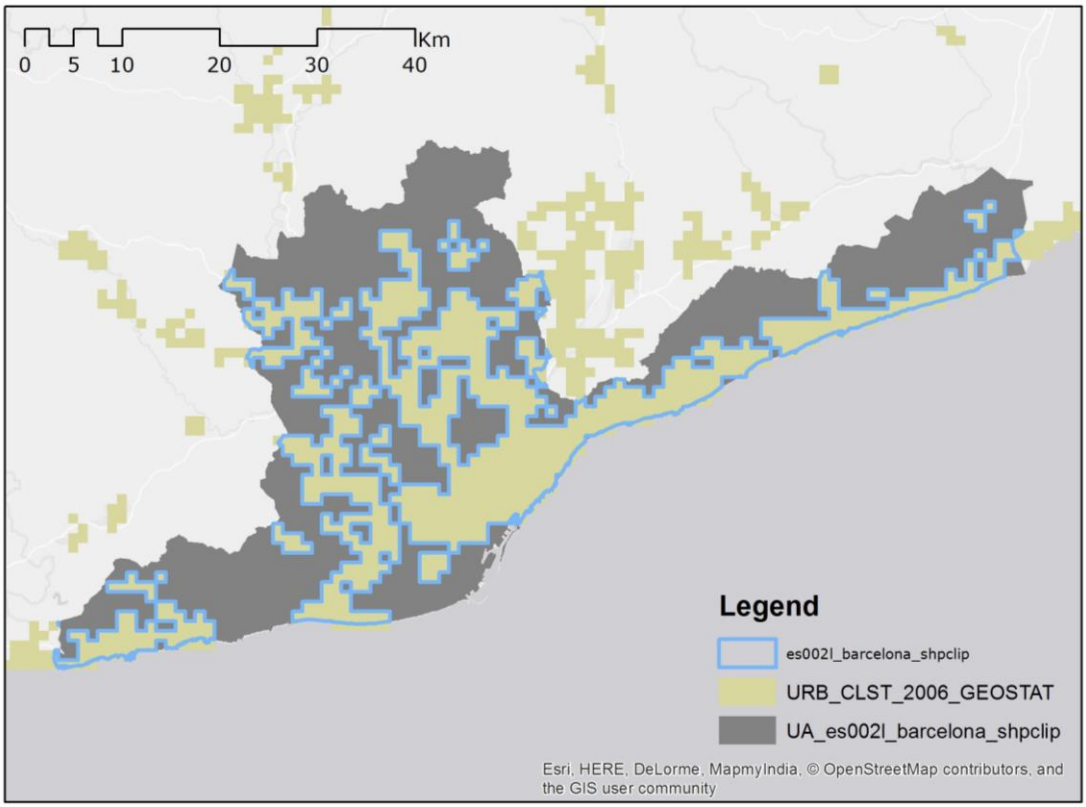
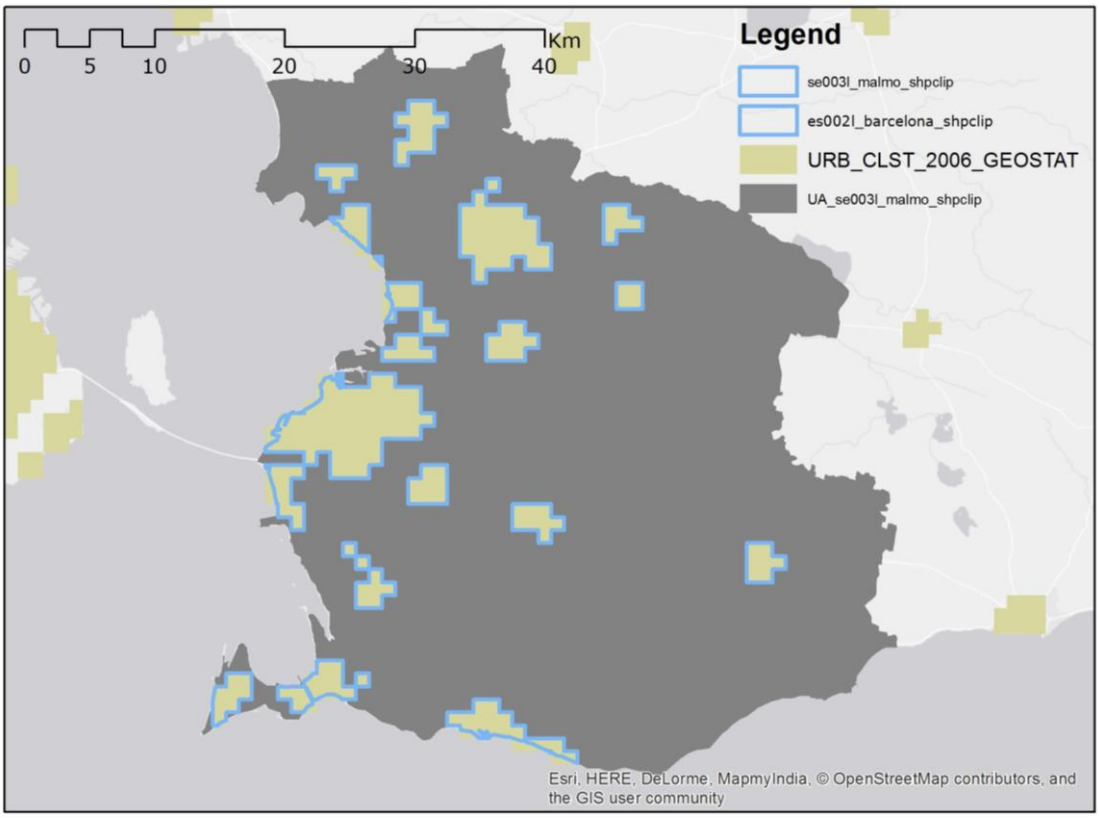


Figure 2 - Difference of city border of UA, Urban cluster and their intersection – City of Malmo, Sweden



2.3 Workflow

The method aims at compare in a proper way the green ESM with the only another dataset which is now available to study urban green areas, the Urban Atlas.

The method includes the definition of the study areas (city masks) as described in the previous paragraph; the preparation and clipping of the input data for the two dataset (one is composed by to raster layers and the other by vector feature class); the calculation of the resident population in the study area suing the zonal statistics on the population grids produced by JRC; and the array throughout the output results to create a final table.

The following paragraphs contain the detailed workflow.

Preparation of the data

The first steps of the workflow are devoted to the preparation of the data and the environment. The creation of a geodatabase is useful in order to avoid hard coding and facilitate the scripting in Python [step 1]. The workspace is defined in the same geodatabase [step 2]. An array containing the list of cities extracted from the name coding of the UA cities is done [step 3].

1. *Create File GDB*
2. *Definition of workshop (geodatabase)*
3. *List of cities.*

Definition of the city masks

The tiff layer containing the urban clusters elaborated by GEOSTAT is converted from raster to polygons [step 4]. The UA feature classes are dissolved to obtain the city boarder polygons of the UA [step 5]. The two different city masks (Geostat and UA) are intersected to obtain the city mask [step 6].

4. *Geoprocess - Conversion of GEOSTAT urban clusters from raster to polygons*
5. *Geoprocess - Dissolve the UA shape files to obtain the UA city boarders.*
6. *Geoprocess - Intersection of the Urban cluster boarders with the UA city boarders.*

Clipping the datasets with the city masks

This part of the workflow is devoted to clipping of input datasets and the raster calculator process on the ESM. The UA feature class 14100 (Green urban areas) is extracted [step 7] and clipped with the city mask [step 8]. The ESM is processed through the following steps: the layers are clipped [steps 9 -10]; the two layers 40 and 41 are merged, with no data value is assigned as 255[5], and the pixels with values bigger than 50 (meaning that more than the half of the pixel is green) are identify as green (value = 1) [step 11]; the raster layer is then converted into single polygons [steps 12-13]; the polygons bigger than 2500 sqm (minimum mapping unit of the UA) are selected [step 14].

7. *Extraction of the UA feature class 14100 – urban green*
8. *Clipping of UA feature class 14100 with the city mask*
9. *Clipping of the ESM layer 40 with the city mask*
10. *Clipping of the ESM layer 40 with the city mask*
11. *Raster calculator to merge the ESM layers 40 and 41*
12. *Conversion of ESM raster to polygons*

[5] The assignment of the correct no data value is necessary to perform correctly the raster calculator.

13. *Conversion of ESM from multipart to single part*
14. *Feature Class to Feature Class*

Calculation of the resident population

In order to calculate the green area per capita it is necessary to extract the resident population in the city mask through a zonal statistic tool. The input data used is the population grip elaborated by the JRC using GEOSTAT population of 2011 (pop_GEOSTAT_2011_V2_JRC_tif) [step 15].

15. *Geoprocess - Calculation of population living within the borders of the city mask - Zonal Statistics as Table"*

Creation of a table with the output results

In order to compare the results, an output table is created with the following fields: 'City', 'AREA', 'pop', 'ESM_GA', 'UA_GA' [step 16].

16. *Geoprocess - Creation of an output table and addition of fields*

Creation of cursor and filling up the output table

17. *Creation of the cursor and filling the output results*

3. Results

The output data shows information about the area of the city, resident population in the study area, the amount of green area detected by the ESM, and the total area of the feature class 14100 of the UA (urban green area).

The chart below shows the relation between the total amount of green ESM and the total amount of the green UA per city (Chart 1- Linear Regression ESM - UAChart 1).

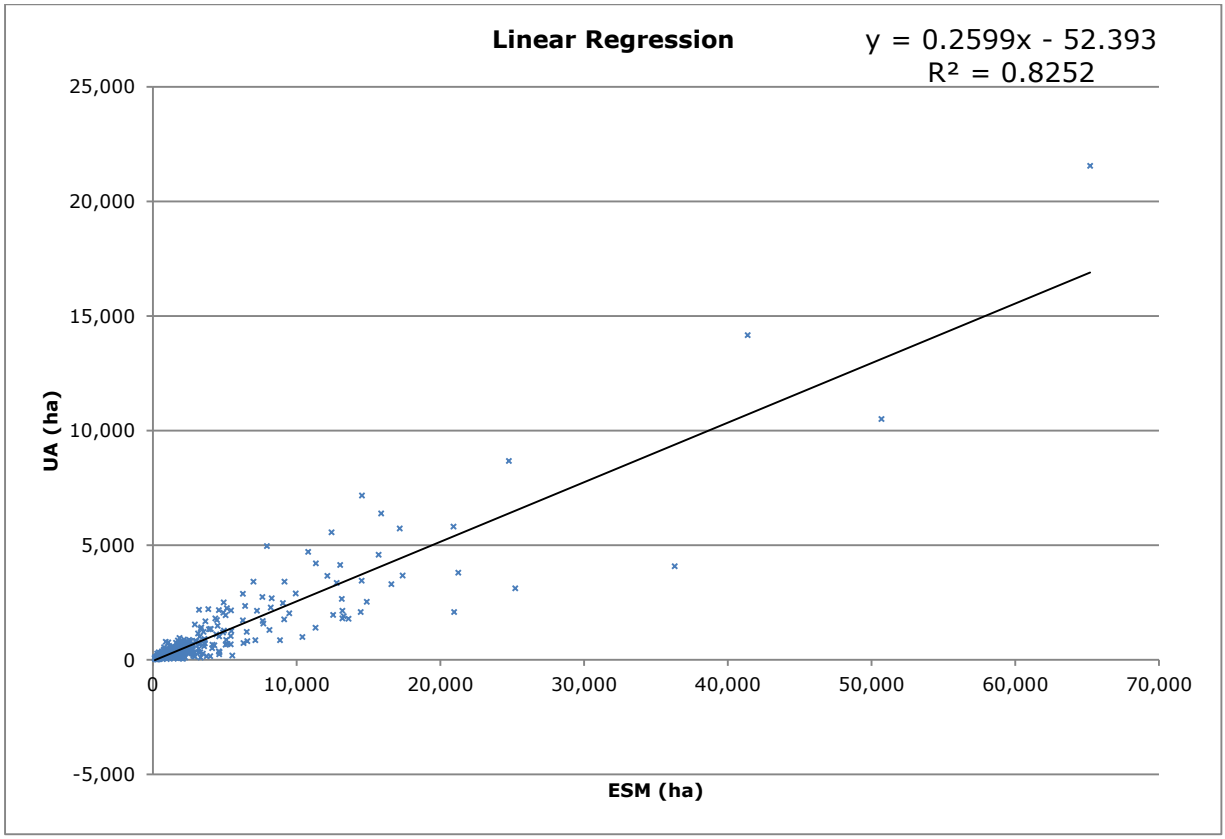
The regression equation is defined as follows:

$$Y = bX + a$$

where UA is the independent variable *X*, green ESM is the dependent variable *Y,* the slope of the line is *b* and the intercept is *a*.

The R-squared[6] value R^2 is 0.8252 which demonstrates that there is a good relation between the two datasets. The slope of the line *b* is around 0.25 meaning that it can be said that the ESM detects four times more urban green areas than the UA.

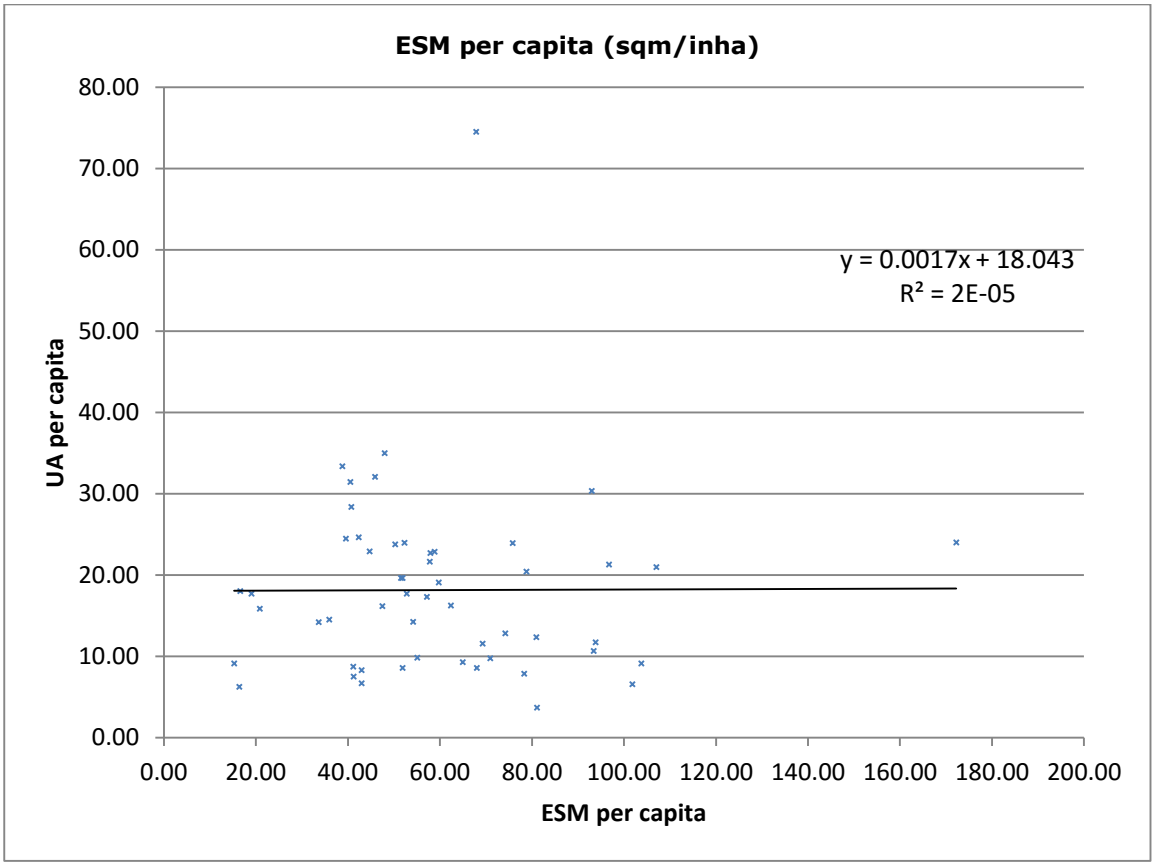
Chart 1- Linear Regression ESM - UA



[6] The R-squared value (a number from 0 to 1 that reveals how closely the estimated values for the trendline correspond to actual data)

If we look at the linear regression of the ESM per capita and the UA per capita this is much lower, due to the fact that we introduced a variable (the resident population), which is not related directly to the green areas.
The chart below shows the linear regression and the R-squared for this data (Chart 2).

Chart 2 - ESM - UA per capita

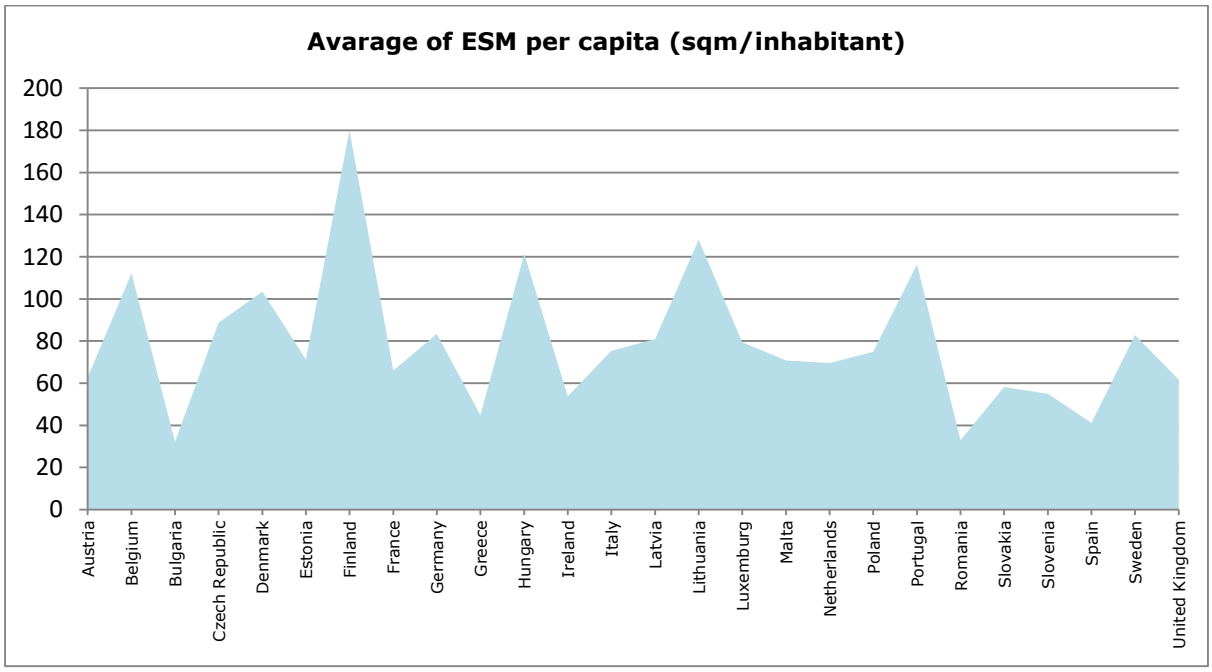


The following table and chart show the average values of the ESM per capita per each of the 26 European countries analysed in the study and the number of cities studied per each country. From this table results that: the European average is 72.53 sqm per inhabitant, the country with the highest average is Finland with 180.07 sqm per inhabitant, and the one with the lowest average value is Bulgaria with 32.17 sqm per inhabitant (Chart 3 and Table 1).

Table 1 - Average of ESM per capita per country

Country	Average of ESM per capita (sqm/inha)	Count of City
Austria	62.67	5
Belgium	112.05	7
Bulgaria	**32.17**	**8**
Czech Republic	88.76	13
Denmark	103.50	5
Estonia	71.21	2
Finland	**180.07**	**4**
France	66.07	28
Germany	83.44	34
Greece	44.74	9
Hungary	121.36	9
Ireland	53.72	5
Italy	75.37	32
Latvia	80.94	2
Lithuania	128.01	3
Luxemburg	79.44	1
Malta	70.74	2
Netherlands	69.60	14
Poland	74.94	27
Portugal	116.35	9
Romania	32.92	14
Slovakia	58.14	8
Slovenia	55.02	2
Spain	41.18	25
Sweden	82.82	8
United Kingdom	61.62	25
Average	**72.53**	
Count		**300**

Chart 3 - Average of ESM per capita per country



The table below shows the results for the 26 EU capitals analysed in the study ordered by ESM per capita (sqm/inha) (Table 2).

Table 2 - Values of green areas in 26 EU capitals

Country	City	AREA (Ha)	Population	ESM (ha)	ESM per capita (sqm/inha)	% of ESM/ Total Area
Finland	Helsinki	51,637	1,149,655	17,370	151.09	34%
Lithuania	Vilnius	31,600	562,392	8,269	147.03	26%
Hungary	Budapest	75,011	2,388,606	25,220	105.58	34%
Denmark	Kobenhavn	68,835	1,654,880	15,711	94.94	23%
Latvia	Riga	35,529	801,569	7,251	90.46	20%
Ireland	Dublin	58,075	1,482,754	13,044	87.97	22%
Estonia	Tallinn	15,190	440,294	3,520	79.95	23%
Luxemburg	Luxembourg	22,892	326,174	2,591	79.44	11%
Sweden	Stockholm	75,333	1,918,730	14,545	75.81	19%
Czech Republic	Praha	70,061	1,667,207	12,431	74.56	18%
Italy	Roma	139,614	3,054,737	21,242	69.54	15%
Poland	Warszawa	81,471	2,394,189	16,592	69.30	20%
Belgium	Bruxelles	74,563	1,858,448	12,784	68.79	17%
Austria	Wien	56,040	2,061,359	12,152	58.95	22%
Germany	Berlin	151,173	4,245,727	24,777	58.36	16%
Netherlands	Sgravenhage	23,403	1,009,314	5,441	53.91	23%
Portugal	Lisboa	62,411	2,371,882	12,550	52.91	20%
United Kingdom	London	344,201	12,469,157	65,213	52.30	19%
Malta	Valletta	12,068	349,979	1,723	49.23	14%
Slovakia	Bratislava	16,255	472,844	1,832	38.74	11%
France	Paris	257,505	11,130,276	41,376	37.17	16%
Slovenia	Ljubljana	20,865	361,968	1,299	35.89	6%
Bulgaria	Sofia	29,135	1,265,148	4,497	35.55	15%
Spain	Madrid	91,842	6,060,507	15,874	26.19	17%
Romania	Bucuresti	36,190	2,089,697	5,458	26.12	15%
Greece	Athina	65,594	3,543,420	9,059	25.57	14%

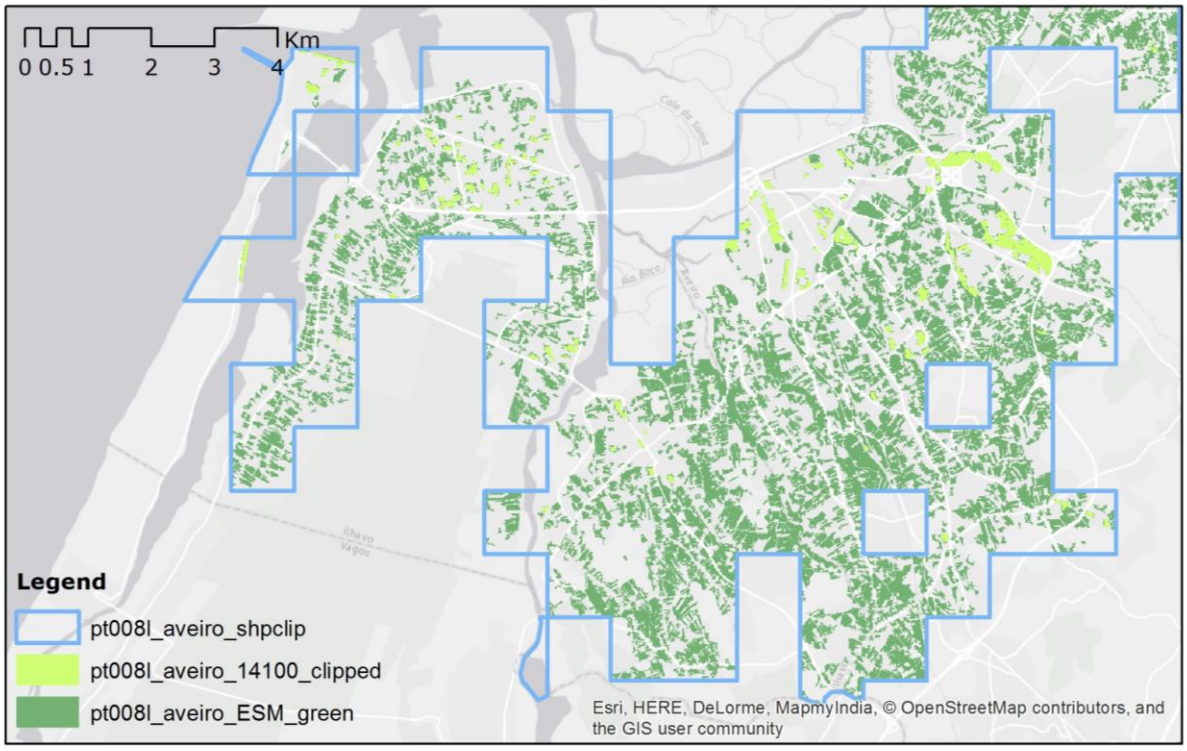
The table below shows the top 10 cities which report the highest values of ESM per capita (Table 3).

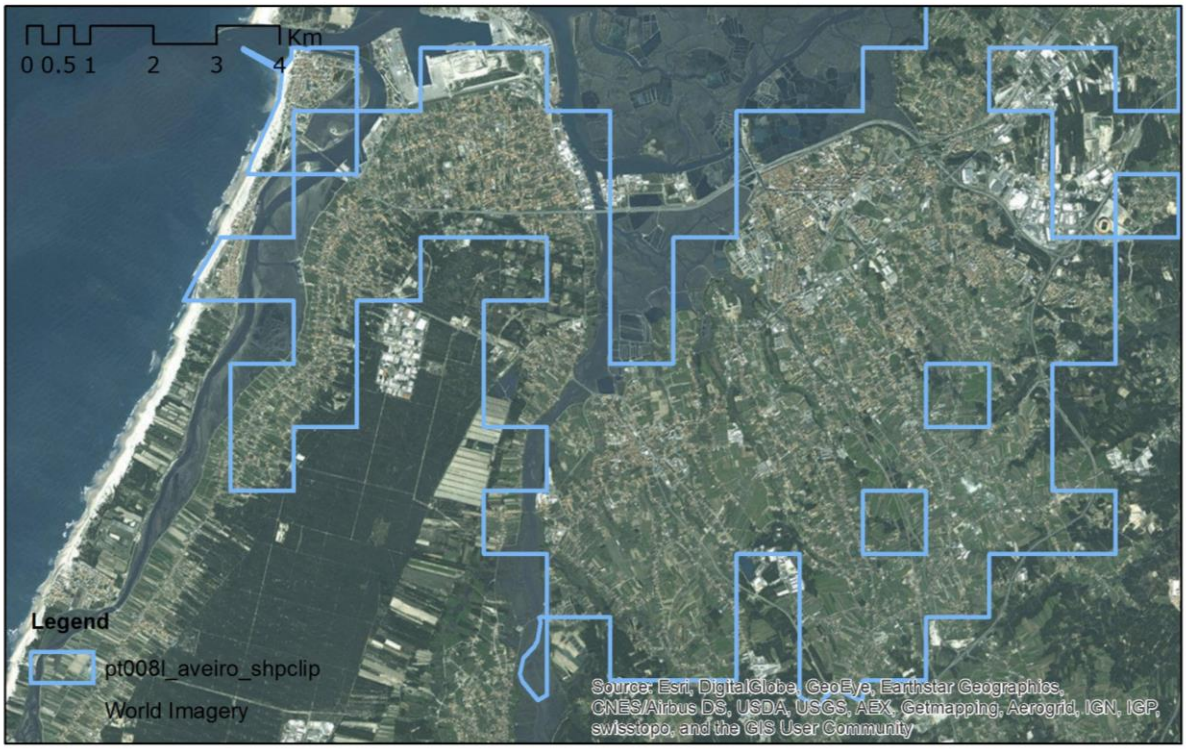
Table 3 - Top 10 cities classified by ESM per capita

Country	City	Area (ha)	Population	ESM (ha)	UA (ha)	ESM per capita (sqm/in ha)	UA per capita (sqm/in ha)	%of ESM/ Total Area
Portugal	Aveiro	9,835	105,861	2,817	170	266.11	16.07	29%
Portugal	Braga	16,459	208,306	4,624	234	221.98	11.23	28%
Finland	Oulu	12,336	153,454	3,360	1,410	218.97	91.91	27%
Finland	Turku	15,208	227,992	4,943	1,274	216.79	55.88	33%
Hungary	Kecskemet	8,767	121,632	2,190	240	180.01	19.69	25%
United Kingdom	Aberdeen	15,141	314,498	5,415	667	172.18	21.21	36%
Belgium	Bruges	8,392	131,522	2,103	315	159.90	23.93	25%
Poland	Poznan	32,500	836,533	13,190	2,137	157.68	25.55	41%
Czech Republic	Ostrava	59,123	844,200	13,141	2,654	155.66	31.44	22%
Finland	Helsinki	51,637	1,149,655	17,370	3,665	151.09	31.88	34%

In the following page, in the Figure 3 are shown the ESM green areas and the UA green areas for the city of Aveiro, Portugal which is the one registering the highest value of ESM per capita.

Figure 3 - City of Aveiro, Portugal – ESM and UA green areas





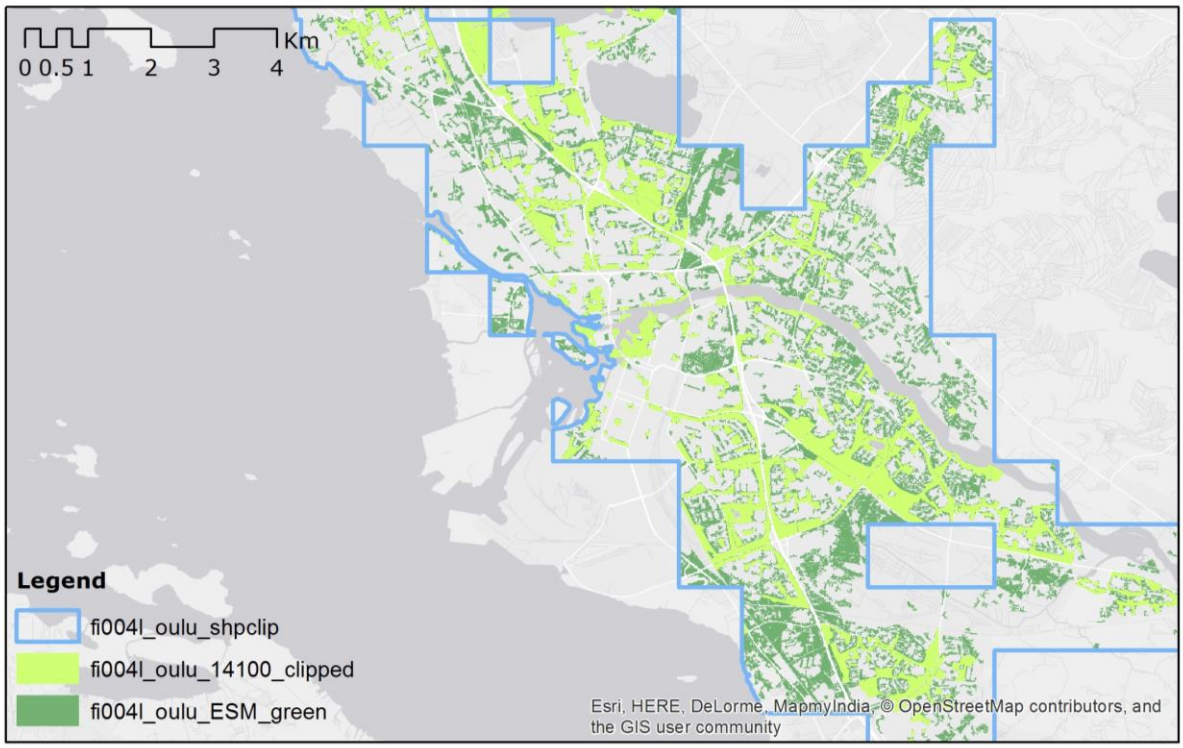
The table below shows the top 10 cities which report the highest values of UA per capita (Table 4 -Top ten cities by UA per capita).

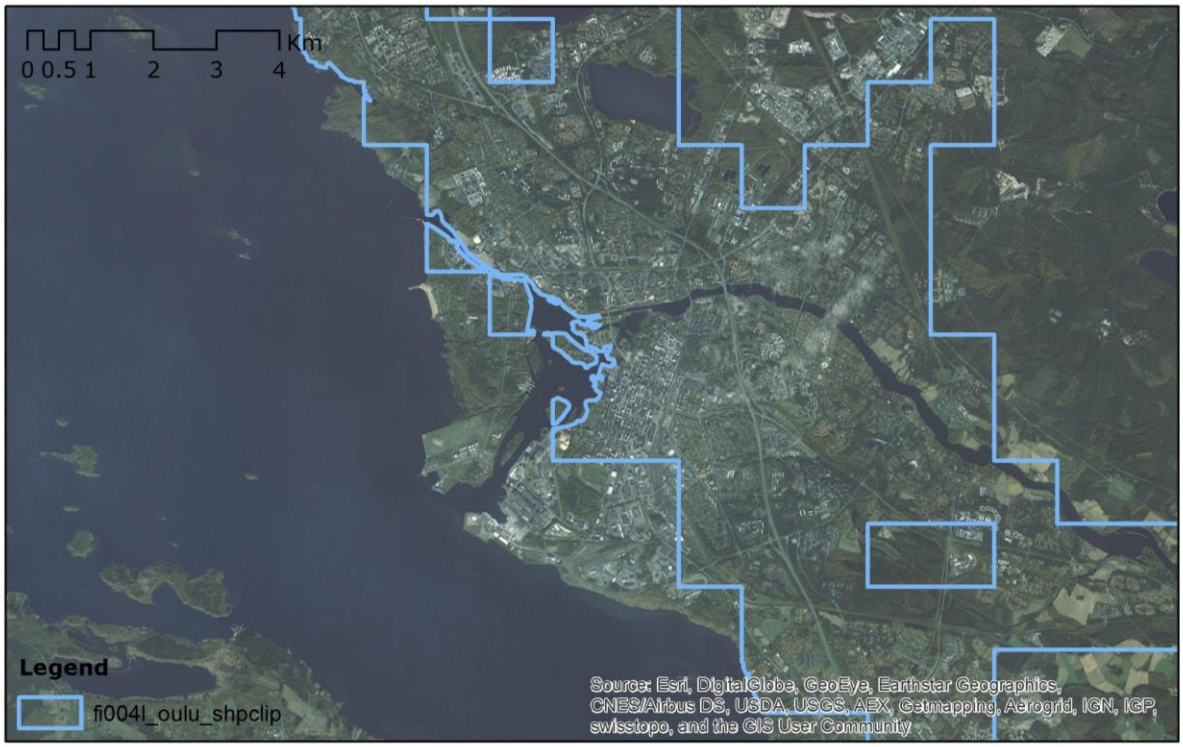
Table 4 -Top ten cities by UA per capita

Country	City	Area (ha)	Population	ESM (ha)	UA (ha)	ESM per capita (sqm/inha)	UA per capita (sqm/inha)	%of ESM/ Total Area
Finland	Oulu	12,336	153,454	3,360	1,410	218.97	91.91	27%
Finland	Tampere	20,200	289,431	3,862	2,204	133.42	76.16	19%
Czech Republic	Karlovyvary	4,674	72,688	790	542	108.72	74.53	17%
Lithuania	Kaunas	16,945	347,215	4,597	2,167	132.39	62.42	27%
Sweden	Orebro	6,700	131,488	894	778	67.97	59.15	13%
Sweden	Linkoping	6,800	131,905	1,069	746	81.05	56.56	16%
Germany	Frankfurt_oder	2,962	52,494	622	297	118.50	56.55	21%
Finland	Turku	15,208	227,992	4,943	1,274	216.79	55.88	33%
Sweden	Jonkoping	8,600	111,468	1,135	604	101.80	54.20	13%
Lithuania	Vilnius	31,600	562,392	8,269	2,676	147.03	47.58	26%

In the following page, in the Figure 3 is showed the ESM green areas and the UA green areas overlapped in the city of Oulu, Finland, the one which registered the highest value of ESM per capita.

Figure 4 - City of Oulu, Finland – ESM and UA green areas





4. Conclusion

The report has demonstrated that the Green ESM is a useful tool to measure the presence of urban green areas in European cities.

Other datasets, such as the UA, do not cover the entire European territory, and they do not allow performing valuable comparative studies.

Instead, the Green ESM allows obtaining information about urban green areas with a high resolution (10m) and potentially in the near future, also at 2.5m of resolution, when the new ESM layers will be available.

The report demonstrates that ESM can be a useful set of information to evaluate the effectiveness of EU and local policies for urban green areas.

The present report illustrates the results of the test performed on 300 European cities, but the same method can be applied to all the European urban clusters.

Another valuable development of the study could be linked to the study of the accessibility of urban green areas.

5. Annexes

This section contains the full Python code performed to calculate the results, the table with the results for all 300 cities and the results aggregated per country, as follows:

- Python code
- Complete Results:
  - Table - Complete results of the 300 cities analysed in the work
  - Table - European cities with more than 1 million of inhabitants ordered by ESM per capita
  - Charts with the results aggregated per country:
    - Austria
    - Belgium
    - Bulgaria
    - Czech Republic
    - Germany
    - Denmark
    - Estonia
    - Spain
    - Finland
    - France
    - Greece
    - Hungary
    - Ireland
    - Italy
    - Lithuania - Luxemburg - Latvia - Malta
    - Netherlands
    - Poland
    - Portugal
    - Romania
    - Sweden
    - Slovenia
    - Slovakia
    - United Kingdom

5.1 Code

```
#Import arcpy module
import arcpy
from arcpy import env
from arcpy.sa import *
# Check out the ArcGIS Spatial Analyst extension license
arcpy.CheckOutExtension("Spatial")
#inport os
import os

   1. Create File GDB
out_folder_path = "C://data/outpath/"
ESMvsUA = "ESMvsUA.gdb"
# Execute CreateFileGDB
arcpy.CreateFileGDB_management(out_folder_path, ESMvsUA)

   2. Definition of workshop (geodatabase)
env.workspace = "C://data/outpath/ESMvsUA.gdb/"

   3. List of cities
#list of cities
UA = []
for root, dirs, files in os.walk(' C://data/EEA_UrbanAtlas/2006'):
        for city in files:
                if city.endswith('.shp'):
                        ua_subelement = []
                        ua_subelement.append(root + "/" + city)
                        ua_subelement.append(city)
                        ua_subelement.append(city.replace(".shp", ""))
                        UA.append(ua_subelement)

   4. Conversion of Geostat urban clusters from raster to polygons
URB_CLST_2006_GEOSTAT_tif = "C://data/URB_CLST_2006_GEOSTAT/URB_CLST_2006_GEOSTAT.tif"
URB_CLST_2006_GEOSTAT_shp = "URB_CLST_2006_GEOSTAT"
# Process: Raster to Polygon
if not(os.path.exists(URB_CLST_2006_GEOSTAT_shp)):
        try:
                arcpy.RasterToPolygon_conversion(URB_CLST_2006_GEOSTAT_tif, URB_CLST_2006_GEOSTAT_shp,
"NO_SIMPLIFY", "VALUE")
        except:
                pass

   5. Dissolve the UA shape files to obtain the UA city boarders
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        outroot = "C://data/outpath/ESMvsUA.gdb/"
        city_ UA_shapeclip = str(city) + "_UA_shpclip"
        # Process: Merge
        if not(os.path.exists(UA_city_shapeclip)):
                try:
                        print UA_city_shapeclip
                        arcpy.Dissolve_management(fullpath, city_ UA_shapeclip, "", "", "SINGLE_PART",
"DISSOLVE_LINES")
                except:
                        print str(city) + "_pass"
                        pass

   6. Intersection of the Urban cluster boarders with the UA city boarders
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        #Local variables
        city_ UA_shapeclip = str(city) + "_UA_shpclip"
        URB_CLST_2006_GEOSTAT_shp = "URB_CLST_2006_GEOSTAT"
        City_shpclip = str(city) + "_shpclip"
        # Process: Intersect
        if not (os.path.exists(City_shpclip)):
                try:
                        print City_shpclip
                        arcpy.Intersect_analysis([city_UA_shapeclip, URB_CLST_2006_GEOSTAT_shp], City_shpclip,
```

```
"ALL", "", "INPUT")
                    except:
                                print str(city) + "_pass"
                                pass
```

7. Extraction of the UA feature class 14100 – urban green

```
for element in UA:
          fullpath = element[0]
          UA_shape = element[1]
          city = element[2]
          outroot = "C://data/outpath/ESMvsUA.gdb/"
          UA_14100 = str(city) + "_14100"
          # Process: Feature Class to Feature Class
          if not(os.path.exists(outroot + UA_14100)):
                    try:
                                print UA_14100
                                arcpy.FeatureClassToFeatureClass_conversion(fullpath, outroot, UA_14100, '"CODE" =
\'14100\'')
                    except:
                                print str(city) + "pass"
                                pass
```

8. Clipping of UA feature class 14100 with the city mask

```
for element in UA:
          #Local variables
          fullpath = element[0]
          UA_shape = element[1]
          city = element[2]
          #Local variables
          UA_14100 = str(city) + "_14100"
          City_shpclip = str(city) + "_shpclip"
          UA_14100_clipped = str(city) + "_14100_clipped"
          # Process: Clip
          if not (os.path.exists(UA_14100_clipped)):
                    try:
                                print UA_14100_clipped
                                arcpy.Clip_analysis(UA_14100, City_shpclip, UA_14100_clipped, "")
                    except:
                                print str(city) + "_pass"
                                pass
```

9. Clipping of the ESM layer 40 with the city mask

```
for element in UA:
          #Local variables
          fullpath = element[0]
          UA_shape = element[1]
          city = element[2]
          #Local variables
          index10m_40_vrt = "C://data/index10m_40.vrt"
          ESM40_clipped_tif = " C://data/ESM/" +str(city) + "_ESM40_clipped.tif"
          City_shpclip = str(city) + "_shpclip"
          # Process: Clip
          if not (os.path.exists(ESM40_clipped_tif)):
                    try:
                                print ESM40_clipped_tif
                                # Process: Clip
                                arcpy.Clip_management(index10m_40_vrt, "#", ESM40_clipped_tif, City_shpclip, "#",
"ClippingGeometry", "NO_MAINTAIN_EXTENT")
                                rasterObj=arcpy.Raster(ESM40_clipped_tif)
                                arcpy.SetRasterProperties_management(rasterObj,nodata="1 255")
                                rasterObj.save(ESM40_clipped_tif)
                    except:
                                print str(city) + "_pass"
                                pass
```

10. Clipping of the ESM layer 40 with the city mask

```
for element in UA:
          #Local variables
          fullpath = element[0]
          UA_shape = element[1]
          city = element[2]
          #Local variables
          index10m_40_vrt = " C://data/ index10m_41.vrt"
          ESM40_clipped_tif = C://data/ESM/" + str(city) + "_ESM41_clipped.tif"
          City_shpclip = str(city) + "_shpclip"

          # Process: Clip
          if not (os.path.exists(ESM41_clipped_tif)):
                    try:
                                print ESM41_clipped_tif
                                # # Process: Clip
```

```
                        arcpy.Clip_management(index10m_41_vrt, "#", ESM41_clipped_tif, City_shpclip, "#",
"ClippingGeometry", "NO_MAINTAIN_EXTENT")
                        rasterObj=arcpy.Raster(ESM41_clipped_tif)
                        arcpy.SetRasterProperties_management(rasterObj,nodata="1 255")
                        rasterObj.save(ESM41_clipped_tif)
                except:
                        print str(city) + "_pass"
                        pass
```

11. Raster calculator to merge the ESM layers 40 and 41

```
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        #Local variables
        env.workspace = "C://data/ESM/"
        ESM40_clipped_tif = str(city) + "_ESM40_clipped.tif"
        ESM41_clipped_tif = str(city) + "_ESM41_clipped.tif"
        ESM40_41_clipped_tif = str(city) + "_ESM40_41_clipped.tif"

        # Process: Clip
        if not (os.path.exists(ESM40_41_clipped_tif)):
                try:
                        print ESM40_41_clipped_tif
                        # Process: Raster Calculator
                        arcpy.gp.RasterCalculator_sa("Con((\""+ESM40_clipped_tif+"\" + \""+ESM41_clipped_tif+"\")>
50,1, 0)", ESM40_41_clipped_tif)
                except:
                        print str(city) + "_pass"
                        pass
```

12. Conversion of ESM raster to polygons

```
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        #Local variables
        ESM40_41_clipped_tif = "C://data/ESM/" + str(city) + "_ESM40_41_clipped.tif"
        ESM_poly = "C://data/ESMvsUA.gdb/" + str(city) + "_ESM_40_41"
        field = "VALUE"

        # Process: Clip
        if not (os.path.exists(ESM_poly)):
                try:
                        print ESM_poly
                        # Process: Raster to Polygon
                        arcpy.RasterToPolygon_conversion(ESM40_41_clipped_tif, ESM_poly, "NO_SIMPLIFY", field)
                except:
                        print str(city) + "_pass"
                        pass
```

13. Conversion of ESM from multipart to single part

```
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        #Local variables

        ESM_poly = "C://data/ESMvsUA.gdb/" + str(city) + "_ESM_40_41"
        ESM_poly_single = "C://data/ESMvsUA.gdb/" + str(city) + "_ESM_40_41_single"

        # Process: mulitpart to single
        if not (os.path.exists(ESM_poly_single)):
                try:
                        print ESM_poly_single
                        # Process: Multipart To Singlepart
                        arcpy.MultipartToSinglepart_management(ESM_poly, ESM_poly_single)
                except:
                        print str(city) + "_pass"
                        pass
```

14. Feature Class to Feature Class

```
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        #Local variables
```

```
        ESM_poly_single = "C://data/ESMvsUA.gdb/" + str(city) + "_ESM_40_41_single"
        out_path = "C://data/ESMvsUA.gdb/"
        ESM_green = str(city) + "_ESM_green"

        if not (os.path.exists(ESM_green)):
                try:
                        print ESM_green
                        # Process: Feature Class to Feature Class
                        arcpy.FeatureClassToFeatureClass_conversion(ESM_poly_single, out_path, ESM_green,
"Shape_Area>2500 AND gridcode=1")
                        print ESM_green + "_done"
                except:
                        print str(city) + "_pass"
                        pass
```

15. Calculation of population living within the borders of the city mask - Zonal Statistics as Table"

```
for element in UA:
        #Local variables
        fullpath = element[0]
        UA_shape = element[1]
        city = element[2]
        #Local variables
        City_shpclip = "C://data/ESMvsUA.gdb/" + str(city) + "_shpclip"
        pop_GEOSTAT_2011_V2_JRC_tif = ""C://data/pop_GEOSTAT_2011_V2_JRC.tif"
        output_zsta_pop = "C://data/ESMvsUA.gdb/" + str(city) + "_pop"

        # Process: ZonalStatisticsAsTable
        if not (os.path.exists(output_zsta_pop)):
                try:
                        print str(city)
                        ZonalStatisticsAsTable(City_shpclip, "OBJECTID", pop_GEOSTAT_2011_V2_JRC_tif,
output_zsta_pop, "DATA", "SUM")
                        print str(city) + "_pop_done"
                except:
                        print str(city) + "empty"
                        pass
```

16. **Creation of an output table and related fields**

```
out_path = "C://data/ESMvsUA.gdb/"
output_table = "output_table"
# Execute CreateTable
arcpy.CreateTable_management(out_path, output_table)
arcpy.AddField_management(output_table, 'City', 'TEXT')
arcpy.AddField_management(output_table, 'AREA', 'DOUBLE')
arcpy.AddField_management(output_table, 'pop', 'DOUBLE')
arcpy.AddField_management(output_table, 'ESM_GA', 'DOUBLE')
arcpy.AddField_management(output_table, 'UA_GA', 'DOUBLE')

#local variables
cursor_final_statistics = arcpy.da.InsertCursor(output_table,['City','AREA','pop', 'ESM_GA','UA_GA'])
```

17. Creation of the cursor and filling the output results

```
for element in UA:
        city = element[2]
        City = city
        ESM_green = "C://data/ESMvsUA.gdb/" + str(city) + "_ESM_green"
        City_shpclip = "C://data/ESMvsUA.gdb/" + str(city) + "_shpclip"
        UA_14100_clipped = "C://data/ESMvsUA.gdb/"  + str(city) + "_14100_clipped"
        output_zsta_pop = "C://data/ESMvsUA.gdb/" + str(city) + "_pop"

        if ((arcpy.Exists(ESM_green)) and (arcpy.Exists(City_shpclip)) and (arcpy.Exists(UA_14100_clipped)) and
(arcpy.Exists(output_zsta_pop))):
                try:
                        table_zsta = "C://data/ESMvsUA.gdb/" + str(city) + "_shpclip"
                        cursor = arcpy.da.SearchCursor(table_zsta, ['Shape_Area'])
                        City_Area = 0
                        for row in cursor:
                                print (row)
                                City_Area = City_Area + (row[0])

                        table_zsta = "C://data/ESMvsUA.gdb/" + str(city) + "_pop"
                        cursor = arcpy.da.SearchCursor(table_zsta, ['SUM'])
                        population = 0
                        for row in cursor:
                                print (row)
                                population = population +(row[0])

                        table_zsta = "C://data/ESMvsUA.gdb/"  + str(city) + "_ESM_green"
                        cursor = arcpy.da.SearchCursor(table_zsta, ['Shape_Area'])
                        ESM_green_area = 0
                        for row in cursor:
```

```
                                        print (row)
                                        ESM_green_area = ESM_green_area + (row[0])

                                table_zsta = "C://data/ESMvsUA.gdb/" + str(city) + "_14100_clipped"
                                cursor = arcpy.da.SearchCursor(table_zsta, ['Shape_Area'])
                                UA_14100_clipped = 0
                                for row in cursor:
                                        print (row[0])
                                        UA_14100_clipped = UA_14100_clipped +(row[0])


                                cursor_final_statistics.insertRow((City, City_Area, population, ESM_green_area,
UA_14100_clipped))
                                # Delete cursor object
                                del cursor
                        except:
                                print str(city) + "pass"
                                pass

del cursor_final_statistics
```

5.2 Complete Results

Table 5 - Complete results of the 300 cities analysed in the work

	Country	City	Area (ha)	pop	ESM (ha)	UA (ha)	ESM per capita (sqm/inha)	UA per capita (sqm/in ha)
1	Austria	Wien	56,040	2,061,359	12,152	3,651	58.95	17.71
2	Austria	Graz	16,300	313,642	1,812	286	57.79	9.11
3	Austria	Linz	22,006	392,078	3,169	769	80.82	19.62
4	Austria	Salzburg	16,150	243,633	1,821	519	74.76	21.29
5	Austria	Innsbruck	12,100	231,032	948	144	41.02	6.24
6	Belgium	Bruxelles	74,563	1,858,448	12,784	3,348	68.79	18.01
7	Belgium	Antwerpen	40,684	913,978	8,120	1,297	88.84	14.19
8	Belgium	Gent	27,612	389,301	5,193	688	133.38	17.68
9	Belgium	Charleroi	22,878	338,226	3,562	586	105.32	17.32
10	Belgium	Liege	37,315	505,483	6,566	801	129.90	15.85
11	Belgium	Brugge	8,392	131,522	2,103	315	159.90	23.93
12	Belgium	Namur	7,322	95,575	939	290	98.22	30.33
13	Bulgaria	Sofia	29,135	1,265,148	4,497	1,484	35.54	11.73
14	Bulgaria	Plovdiv	8,198	360,090	623	236	17.30	6.55
15	Bulgaria	Varna	10,476	347,077	2,434	338	70.13	9.74
16	Bulgaria	Burgas	3,974	188,737	257	148	13.62	7.85
17	Bulgaria	Pleven	1,700	101,634	239	37	23.54	3.68
18	Bulgaria	Ruse	2,954	145,327	365	124	25.11	8.56
19	Bulgaria	Vidin	1,654	47,946	138	70	28.71	14.51
20	Bulgaria	Starazagora	2,181	131,112	569	168	43.42	12.80
21	Czech Republic	Praha	70,061	1,667,207	12,431	5,562	74.56	33.36
22	Czech Republic	Brno	23,500	512,155	4,125	503	80.54	9.81
23	Czech Republic	Ostrava	59,123	844,200	13,141	2,654	155.66	31.44
24	Czech Republic	Plzen	9,400	200,344	822	213	41.03	10.63
25	Czech Republic	Usti_nad_labem	12,016	200,966	2,737	456	136.19	22.70
26	Czech Republic	Olomouc	9,100	143,210	1,903	166	132.85	11.57
27	Czech Republic	Liberec	9,802	170,538	1,814	596	106.39	34.97
28	Czech Republic	Ceske_budejovice	5,000	103,338	429	168	41.47	16.25
29	Czech Republic	Hradec_kralove	6,100	100,902	520	242	51.49	23.97
30	Czech Republic	Pardubice	6,100	106,468	622	244	58.40	22.91
31	Czech Republic	Zlin	7,130	108,373	1,166	175	107.55	16.16
32	Czech Republic	Karlovyvary	4,674	72,688	790	542	108.72	74.53
33	Czech Republic	Jihlava	3,000	57,403	339	141	59.01	24.62
34	Germany	Berlin	151,173	4,245,727	24,777	8,663	58.36	20.40
35	Germany	Hamburg	119,675	2,643,543	17,185	5,723	65.01	21.65
36	Germany	Munchen	78,621	2,202,786	11,332	4,207	51.45	19.10
37	Germany	Koln	73,812	1,754,425	14,532	3,444	82.83	19.63
38	Germany	Frankfurt_am_main	104,427	2,247,654	14,455	2,085	64.31	9.28

39	Germany	Frankfurt_am_main_union	162,730	3,118,435	20,965	2,085	67.23	6.69
40	Germany	Stuttgart	118,836	2,395,135	13,185	1,797	55.05	7.50
41	Germany	Leipzig	37,800	699,327	5,155	2,242	73.71	32.05
42	Germany	Dresden	41,555	735,082	9,142	1,763	124.37	23.98
43	Germany	Dusseldorf	61,247	1,446,953	13,607	1,786	94.04	12.34
44	Germany	Bremen	52,082	927,368	8,192	2,268	88.34	24.46
45	Germany	Wuppertal	12,036	335,451	2,876	278	85.74	8.30
46	Germany	Bielefeld	76,132	1,063,149	14,889	2,528	140.05	23.78
47	Germany	Halle_an_der_saale	19,900	319,477	3,587	906	112.29	28.36
48	Germany	Magdeburg	19,292	342,600	2,163	783	63.15	22.85
49	Germany	Wiesbaden	19,128	375,259	2,807	321	74.80	8.55
50	Germany	Gottingen	13,163	205,881	2,799	431	135.97	20.95
51	Germany	Darmstadt	19,189	368,089	1,770	336	48.08	9.13
52	Germany	Trier	9,572	144,746	983	126	67.91	8.72
53	Germany	Freiburg_im_breisgau	29,178	460,542	4,146	655	90.03	14.23
54	Germany	Regensburg	18,594	267,089	1,948	378	72.95	14.14
55	Germany	Frankfurt_oder	2,962	52,494	622	297	118.50	56.55
56	Germany	Weimar	5,700	85,326	682	260	79.98	30.52
57	Germany	Schwerin	11,182	150,879	1,497	455	99.20	30.18
58	Germany	Erfurt	19,100	327,276	2,567	636	78.44	19.44
59	Germany	Augsburg	23,946	479,785	2,365	605	49.30	12.60
60	Germany	Bonn	45,948	790,684	7,132	845	90.20	10.68
61	Germany	Karlsruhe	34,226	627,014	3,197	474	50.99	7.56
62	Germany	Monchengladbach	11,204	262,547	2,803	233	106.77	8.87
63	Germany	Mainz	18,313	350,675	2,090	278	59.59	7.92
64	Germany	Essen	204,048	4,833,300	50,683	10,500	104.86	21.72
65	Germany	Kiel	26,501	442,402	5,116	864	115.64	19.52
66	Germany	Koblenz	16,652	252,808	2,543	363	100.57	14.37
67	Denmark	Kobenhavn	68,835	1,654,880	15,711	4,574	94.94	27.64
68	Denmark	Aarhus	27,430	448,512	4,025	1,337	89.74	29.80
69	Denmark	Aarhus	27,430	448,512	4,025	1,337	89.74	29.80
70	Denmark	Odense	17,809	259,548	3,618	851	139.41	32.78
71	Denmark	Aalborg	17,938	247,032	2,561	764	103.66	30.93
72	Estonia	Tallinn	15,190	440,294	3,520	1,217	79.94	27.65
73	Estonia	Tartu	4,000	102,696	642	233	62.48	22.68
74	Spain	Madrid	91,842	6,060,507	15,874	6,379	26.19	10.53
75	Spain	Barcelona	71,152	4,432,643	7,642	1,692	17.24	3.82
76	Spain	Valencia	25,199	1,622,384	3,608	717	22.24	4.42
77	Spain	Sevilla	27,400	1,306,233	2,169	883	16.61	6.76
78	Spain	Zaragoza	6,989	663,560	766	447	11.54	6.73
79	Spain	Malaga	13,806	690,228	2,244	372	32.52	5.38
80	Spain	Murcia	14,944	488,262	5,528	179	113.22	3.67
81	Spain	Las_palmas	21,246	573,341	3,985	149	69.50	2.59
82	Spain	Valladolid	7,300	374,595	1,102	411	29.43	10.97
83	Spain	Palma_di_mallorca	16,440	530,708	1,730	696	32.59	13.11

84	Spain	Santiago_de_compostela	7,399	126,799	1,438	230	113.44	18.14
85	Spain	Vitoria_gasteiz	2,300	223,032	529	294	23.74	13.18
86	Spain	Oviedo	6,800	271,469	1,223	233	45.04	8.59
87	Spain	Pamplona_iruna	6,254	327,083	774	421	23.66	12.87
88	Spain	Santander	9,206	239,534	1,554	256	64.88	10.68
89	Spain	Toledo	3,558	106,099	494	113	46.58	10.69
90	Spain	Badajoz	3,106	136,220	312	153	22.91	11.20
91	Spain	Logrono	2,000	157,349	312	139	19.82	8.83
92	Spain	Bilbao	16,310	891,129	2,076	529	23.29	5.94
93	Spain	Bilbao_Project	16,323	876,515	2,076	529	23.68	6.04
94	Spain	Cordoba	5,200	304,534	788	191	25.89	6.28
95	Spain	Alicante	8,659	438,875	1,151	359	26.22	8.19
96	Spain	Vigo	25,327	460,112	6,319	721	137.33	15.66
97	Spain	Gijon	3,722	274,459	352	203	12.83	7.38
98	Spain	Santa_cruz_de_tenerife	18,151	458,298	2,254	100	49.17	2.19
99	Finland	Helsinki	51,637	1,149,655	17,370	3,665	151.09	31.88
100	Finland	Tampere	20,200	289,431	3,862	2,204	133.42	76.16
101	Finland	Turku	15,208	227,992	4,943	1,274	216.79	55.88
102	Finland	Oulu	12,336	153,454	3,360	1,410	218.97	91.91
103	France	Paris	257,505	11,130,276	41,376	14,163	37.17	12.72
104	France	Lyon	64,418	1,555,287	9,484	2,023	60.98	13.01
105	France	Toulouse	45,414	868,375	8,841	843	101.81	9.71
106	France	Strasbourg	21,532	529,408	4,281	622	80.87	11.75
107	France	Bordeaux	41,788	796,805	7,687	1,570	96.48	19.70
108	France	Nantes	26,600	568,759	2,914	1,534	51.23	26.97
109	France	Lille	35,904	1,003,935	4,621	1,011	46.03	10.07
110	France	Montpellier	19,645	432,377	1,962	804	45.39	18.59
111	France	Saint_etienne	15,826	341,956	2,416	676	70.64	19.78
112	France	Le_havre	7,773	226,489	1,112	474	49.10	20.94
113	France	Rennes	15,200	357,946	2,480	472	69.28	13.19
114	France	Amiens	7,000	169,372	790	258	46.63	15.23
115	France	Rouen	16,357	403,248	1,607	737	39.86	18.27
116	France	Nancy	16,852	337,949	2,493	672	73.77	19.88
117	France	Metz	19,383	330,260	2,167	546	65.60	16.52
118	France	Reims	5,900	209,830	1,232	237	58.72	11.30
119	France	Orleans	17,212	287,802	1,977	746	68.71	25.93
120	France	Dijon	8,500	243,174	640	364	26.32	14.98
121	France	Poitiers	6,800	119,124	773	192	64.89	16.14
122	France	Caen	16,319	261,354	1,856	399	71.02	15.27
123	France	Limoges	7,700	165,661	1,470	336	88.73	20.31
124	France	Besancon	8,100	142,014	1,590	181	111.98	12.73
125	France	Grenoble	21,512	465,756	3,272	682	70.26	14.63
126	France	Ajaccio	2,013	60,634	283	17	46.67	2.79
127	France	Toulon	14,266	383,088	2,420	786	63.17	20.53
128	France	Tours	15,100	292,335	2,517	868	86.11	29.70

129	France	Aix_en_provence	18,777	270,351	3,300	314	122.06	11.60
130	France	Marseille	24,803	1,002,047	3,667	1,679	36.60	16.75
131	Greece	Athina	65,594	3,543,420	9,059	2,469	25.56	6.97
132	Greece	Thessaloniki	16,999	870,913	2,116	265	24.30	3.05
133	Greece	Patrai	8,483	194,658	2,129	208	109.36	10.71
134	Greece	Iraklion	4,866	157,458	956	29	60.74	1.82
135	Greece	Larisa	4,400	150,145	201	89	13.37	5.95
136	Greece	Volos	3,946	120,072	133	47	11.09	3.95
137	Greece	Ioannina	5,100	94,611	360	86	38.10	9.11
138	Greece	Kavala	1,647	50,037	166	28	33.25	5.59
139	Greece	Kalamata	3,428	56,773	493	37	86.92	6.48
140	Hungary	Budapest	85,011	2,388,606	25,220	3,108	105.59	13.01
141	Hungary	Miskolc	12,166	212,568	2,400	282	112.92	13.26
142	Hungary	Nyiregyhaza	9,706	135,100	1,530	86	113.22	6.37
143	Hungary	Pecs	7,700	158,346	1,907	82	120.46	5.15
144	Hungary	Debrecen	13,288	248,981	2,789	263	112.03	10.56
145	Hungary	Szeged	7,000	167,542	1,297	456	77.40	27.20
146	Hungary	Gyor	8,200	132,365	1,724	289	130.22	21.85
147	Hungary	Kecskemet	8,767	121,632	2,190	240	180.01	19.69
148	Hungary	Szekesfehervar	5,110	103,116	1,448	92	140.40	8.92
149	Ireland	Dublin	58,075	1,482,754	13,044	4,132	87.97	27.87
150	Ireland	Cork	15,317	247,156	730	287	29.55	11.62
151	Ireland	Limerick	9,300	131,046	602	82	45.93	6.22
152	Ireland	Galway	4,087	69,026	498	182	72.09	26.32
153	Ireland	Waterford	2,218	43,785	145	103	33.08	23.51
154	Italy	Roma	139,614	3,054,737	21,242	3,799	69.54	12.44
155	Italy	Milano	78,925	2,641,040	9,948	2,885	37.67	10.92
156	Italy	Napoli	47,305	2,110,311	10,408	993	49.32	4.71
157	Italy	Torino	52,349	1,488,264	6,254	1,712	42.02	11.51
158	Italy	Palermo	24,522	830,226	3,184	331	38.35	3.99
159	Italy	Genova	17,631	512,004	1,624	213	31.72	4.16
160	Italy	Firenze	24,848	552,116	2,819	671	51.06	12.16
161	Italy	Bari	21,956	478,945	1,780	260	37.17	5.43
162	Italy	Bologna	20,470	478,119	2,601	820	54.41	17.14
163	Italy	Catania	20,234	516,937	2,921	138	56.51	2.66
164	Italy	Venezia	32,356	343,079	3,211	684	93.59	19.93
165	Italy	Verona	28,470	381,413	4,590	360	120.33	9.43
166	Italy	Cremona	5,300	69,288	444	98	64.04	14.12
167	Italy	Trento	10,899	111,876	1,106	67	98.82	6.01
168	Italy	Trieste	7,625	152,669	1,064	173	69.70	11.32
169	Italy	Perugia	17,784	136,886	1,499	130	109.54	9.46
170	Italy	Ancona	10,109	138,284	973	198	70.39	14.32
171	Italy	Laquila	4,200	38,754	266	53	68.61	13.79
172	Italy	Pescara	18,339	265,003	3,751	146	141.54	5.50
173	Italy	Campobasso	4,400	50,413	480	31	95.27	6.22
174	Italy	Caserta	19,146	332,006	2,161	156	65.10	4.70

175	Italy	Taranto	21,388	325,470	2,686	303	82.54	9.31
176	Italy	Potenza	6,600	64,553	671	38	103.97	5.88
177	Italy	Catanzaro	5,705	87,674	429	23	48.97	2.57
178	Italy	Reggio_di_calabria	10,752	190,119	2,041	50	107.34	2.64
179	Italy	Sassari	12,963	159,793	2,089	45	130.75	2.83
180	Italy	Cagliari	17,714	423,431	1,309	261	30.90	6.17
181	Italy	Padova	33,858	405,032	4,646	332	114.70	8.20
182	Italy	Brescia	21,879	326,654	3,481	267	106.57	8.18
183	Italy	Modena	11,637	210,512	1,450	424	68.87	20.16
184	Italy	Foggia	5,800	148,359	225	79	15.13	5.30
185	Italy	Salerno	22,099	242,248	3,330	91	137.48	3.75
186	Lithuania	Vilnius	31,600	562,392	8,269	2,676	147.03	47.58
187	Lithuania	Kaunas	16,945	347,215	4,597	2,167	132.39	62.42
188	Lithuania	Panevezys	4,000	103,327	1,081	376	104.63	36.41
189	Luxemburg	Luxembourg	22,892	326,174	2,591	383	79.44	11.76
190	Latvia	Riga	35,529	801,569	7,251	2,134	90.46	26.62
191	Latvia	Liepaja	4,127	80,113	572	209	71.42	26.13
192	Malta	Valletta	12,068	349,979	1,723	108	49.23	3.09
193	Malta	Gozo	2,506	24,872	229	22	92.24	8.83
194	Netherlands	Sgravenhage	23,403	1,009,314	5,441	2,145	53.90	21.25
195	Netherlands	Amsterdam	37,707	1,486,031	6,259	2,876	42.12	19.35
196	Netherlands	Rotterdam	31,042	1,160,760	6,417	2,338	55.28	20.14
197	Netherlands	Utrecht	15,709	571,544	3,258	1,007	57.00	17.62
198	Netherlands	Eindhoven	14,886	402,107	3,004	826	74.70	20.55
199	Netherlands	Tilburg	10,029	275,011	1,899	542	69.06	19.69
200	Netherlands	Groningen	11,894	276,508	1,861	946	67.30	34.21
201	Netherlands	Enschede	11,073	295,302	1,670	554	56.55	18.78
202	Netherlands	Arnhem	14,651	325,792	2,937	807	90.16	24.77
203	Netherlands	Heerlen	11,365	244,217	2,711	486	111.01	19.88
204	Netherlands	Breda	11,590	278,298	2,313	513	83.12	18.44
205	Netherlands	Nijmegen	11,169	260,701	2,143	702	82.18	26.92
206	Netherlands	Apeldoorn	6,876	174,883	1,702	443	97.35	25.31
207	Netherlands	Leeuwarden	5,670	118,551	411	231	34.69	19.48
208	Poland	Warszawa	81,471	2,394,189	16,592	3,290	69.30	13.74
209	Poland	Lodz	30,221	966,135	6,521	1,217	67.50	12.59
210	Poland	Krakow	48,363	956,691	13,268	1,892	138.69	19.78
211	Poland	Wroclaw	25,800	788,669	4,875	1,190	61.81	15.08
212	Poland	Poznan	32,500	836,533	13,190	2,137	157.68	25.55
213	Poland	Gdansk	30,130	919,230	5,412	1,033	58.88	11.24
214	Poland	Szczecin	18,500	593,436	1,739	835	29.31	14.08
215	Poland	Bydgoszcz	10,009	408,102	2,006	314	49.15	7.69
216	Poland	Lublin	13,472	442,559	2,515	717	56.84	16.20
217	Poland	Katowice	100,350	2,459,868	36,300	4,071	147.57	16.55
218	Poland	Bialystok	12,000	383,940	1,622	393	42.25	10.24
219	Poland	Kielce	9,400	213,214	2,025	208	94.99	9.76
220	Poland	Torun	7,707	231,331	2,007	467	86.77	20.21

221	Poland	Olsztyn	6,700	207,959	779	407	37.48	19.59
222	Poland	Rzeszow	12,400	206,085	1,972	198	95.68	9.59
223	Poland	Opole	8,100	149,021	512	171	34.34	11.49
224	Poland	Gorzow_wielkopolski	4,877	142,855	846	399	59.24	27.95
225	Poland	Zielona_gora	5,300	143,223	576	195	40.24	13.58
226	Poland	Jelenia_gora	5,911	95,327	1,032	148	108.27	15.47
227	Poland	Nowy_sacz	9,785	116,123	1,450	44	124.88	3.83
228	Poland	Suwalki	2,100	67,123	145	35	21.61	5.17
229	Poland	Konin	4,100	79,281	702	136	88.57	17.18
230	Poland	Czestochowa	14,519	270,904	2,590	342	95.60	12.61
231	Poland	Radom	10,600	246,679	1,999	202	81.03	8.20
232	Poland	Plock	3,800	122,449	620	190	50.66	15.52
233	Poland	Kalisz	11,200	214,582	2,113	180	98.45	8.38
234	Poland	Koszalin	3,400	113,598	301	258	26.49	22.72
235	Portugal	Lisboa	62,411	2,371,882	12,550	1,947	52.91	8.21
236	Portugal	Oporto	43,362	1,090,562	11,327	1,400	103.86	12.83
237	Portugal	Braga	16,459	208,306	4,624	234	221.98	11.23
238	Portugal	Funchal	9,920	191,661	1,902	64	99.23	3.36
239	Portugal	Coimbra	11,671	139,421	1,950	264	139.86	18.96
240	Portugal	Setubal	4,495	113,579	746	127	65.64	11.20
241	Portugal	Ponta_delgada	5,455	74,933	419	36	55.96	4.78
242	Portugal	Aveiro	9,835	105,861	2,817	170	266.11	16.07
243	Portugal	Faro	4,914	92,359	384	39	41.57	4.27
244	Romania	Bucuresti	36,190	2,089,697	5,458	1,248	26.12	5.97
245	Romania	Cluj_napoca	7,989	294,692	543	120	18.44	4.08
246	Romania	Timisoara	7,594	332,542	732	201	22.01	6.05
247	Romania	Craiova	5,379	262,170	371	203	14.14	7.74
248	Romania	Braila	2,962	175,442	334	76	19.03	4.31
249	Romania	Oradea	5,436	193,589	296	111	15.29	5.75
250	Romania	Bacau	5,418	152,155	789	105	51.87	6.91
251	Romania	Arad	5,800	159,855	1,546	146	96.70	9.15
252	Romania	Sibiu	4,495	164,842	270	89	16.38	5.37
253	Romania	Targu_mures	5,701	156,790	260	30	16.59	1.89
254	Romania	Piatra_neamt	4,955	92,006	309	11	33.62	1.21
255	Romania	Calarasi	2,042	71,154	375	22	52.76	3.13
256	Romania	Giurgiu	1,546	57,886	331	29	57.11	4.98
257	Romania	Alba_iulia	4,273	83,347	174	82	20.84	9.89
258	Sweden	Stockholm	75,333	1,918,730	14,545	7,167	75.80	37.35
259	Sweden	Goteborg	40,597	751,568	6,989	3,401	93.00	45.26
260	Sweden	Malmo	22,001	524,325	4,918	2,492	93.80	47.53
261	Sweden	Jonkoping	8,600	111,468	1,135	604	101.80	54.20
262	Sweden	Umea	4,802	86,945	616	257	70.89	29.56
263	Sweden	Uppsala	6,600	163,588	1,281	556	78.28	33.99
264	Sweden	Linkoping	6,800	131,905	1,069	746	81.05	56.56
265	Sweden	Orebro	6,700	131,488	894	778	67.97	59.15
266	Slovenia	Ljubljana	20,865	361,968	1,299	331	35.89	9.15

267	Slovenia	Maribor	12,500	152,026	1,127	108	74.15	7.08
268	Slovakia	Bratislava	16,255	472,844	1,832	519	38.75	10.98
269	Slovakia	Kosice	7,200	253,745	1,397	318	55.05	12.52
270	Slovakia	Banska_bystrica	4,600	83,824	339	137	40.48	16.39
271	Slovakia	Nitra	6,326	96,029	897	269	93.42	28.02
272	Slovakia	Presov	4,400	98,038	568	137	57.91	13.95
273	Slovakia	Zilina	6,904	107,601	745	124	69.23	11.52
274	Slovakia	Trnava	2,600	69,516	333	79	47.96	11.32
275	Slovakia	Trencin	4,601	70,603	440	56	62.34	7.89
276	United Kingdom	London	344,201	12,469,157	65,213	21,540	52.30	17.28
277	United Kingdom	Birmingham	76,355	2,418,183	10,812	4,708	44.71	19.47
278	United Kingdom	Glasgow	73,810	1,672,626	7,935	4,954	47.44	29.62
279	United Kingdom	Liverpool	44,062	1,351,158	9,168	3,408	67.85	25.22
280	United Kingdom	Edinburgh	32,225	761,305	3,221	2,181	42.30	28.64
281	United Kingdom	Manchester	87,822	2,655,249	20,909	5,807	78.75	21.87
282	United Kingdom	Cardiff	41,166	847,659	4,896	2,054	57.76	24.23
283	United Kingdom	Sheffield	57,921	1,278,702	7,632	2,731	59.69	21.36
284	United Kingdom	Bristol	37,365	983,568	5,063	1,943	51.48	19.75
285	United Kingdom	Belfast	26,806	600,187	3,896	1,330	64.91	22.17
286	United Kingdom	Newcastle_upon_tyne	38,776	1,010,692	4,338	1,801	42.92	17.82
287	United Kingdom	Leicester	29,247	763,721	3,148	1,144	41.22	14.98
288	United Kingdom	Derry	5,059	88,915	408	325	45.88	36.54
289	United Kingdom	Aberdeen	15,141	314,498	5,415	667	172.18	21.21
290	United Kingdom	Cambridge	8,203	178,514	1,444	513	80.92	28.72
291	United Kingdom	Exeter	16,146	301,207	1,191	242	39.55	8.04
292	United Kingdom	Lincoln	7,800	142,294	611	287	42.97	20.19
293	United Kingdom	Wrexham	15,437	202,803	1,019	219	50.25	10.81
294	United Kingdom	Portsmouth	15,355	479,012	1,949	585	40.70	12.22
295	United Kingdom	Worcester	10,160	202,574	1,191	369	58.79	18.20
296	United Kingdom	Coventry	23,950	640,379	3,323	1,337	51.89	20.88
297	United Kingdom	Kingston_upon_hull	19,439	472,768	5,061	657	107.05	13.91
298	United Kingdom	Stoke_on_trent	19,522	429,679	4,458	1,736	103.75	40.40
299	United Kingdom	Wolverhampton	13,898	344,650	1,418	558	41.14	16.18
300	United Kingdom	Nottingham	30,807	817,946	4,427	1,098	54.12	13.43
	Average		**24,649**	**655,977**	**4,217**	**1,043**	**72.55**	**16.64**

4.3 Results for cities with more than 1 million of inhabitants

Table 6 - European cities with more than 1 million of inhabitants ordered by ESM per capita (sqm/inhabitants)

Country	City	Area (ha)	AREA	pop	ESM per capita (sqm/inha)	UA per capita (sqm/inha)	%of ESM/Total Area
Finland	Helsinki	51,637	516,366,521	1,149,655	151.09	31.88	34%
Poland	Katowice	100,350	1,003,498,341	2,459,868	147.57	16.55	36%
Germany	Bielefeld	76,132	761,317,732	1,063,149	140.05	23.78	20%
Hungary	Budapest	85,011	850,113,254	2,388,606	105.59	13.01	30%
Germany	Essen	204,048	2,040,478,402	4,833,300	104.86	21.72	25%
Portugal	Oporto	43,362	433,615,694	1,090,562	103.86	12.83	26%
Denmark	Kobenhavn	68,835	688,346,495	1,654,880	94.94	27.64	23%
Germany	Dusseldorf	61,247	612,468,733	1,446,953	94.04	12.34	22%
Ireland	Dublin	58,075	580,747,409	1,482,754	87.97	27.87	22%
Germany	Koln	73,812	738,124,903	1,754,425	82.83	19.63	20%
United Kingdom	Manchester	87,822	878,215,001	2,655,249	78.75	21.87	24%
Sweden	Stockholm	75,333	753,332,717	1,918,730	75.80	37.35	19%
Czech Republic	Praha	70,061	700,613,611	1,667,207	74.56	33.36	18%
Italy	Roma	139,614	1,396,140,647	3,054,737	69.54	12.44	15%
Poland	Warszawa	81,471	814,709,465	2,394,189	69.30	13.74	20%
Belgium	Bruxelles	74,563	745,632,029	1,858,448	68.79	18.01	17%
United Kingdom	Liverpool	44,062	440,620,676	1,351,158	67.85	25.22	21%
Germany	Frankfurt Am Main Union	162,730	1,627,302,224	3,118,435	67.23	6.69	13%
Germany	Hamburg	119,675	1,196,751,603	2,643,543	65.01	21.65	14%
Germany	Frankfurt Am Main	104,427	1,044,272,263	2,247,654	64.31	9.28	14%
France	Lyon	64,418	644,183,703	1,555,287	60.98	13.01	15%
United Kingdom	Sheffield	57,921	579,209,235	1,278,702	59.69	21.36	13%
Austria	Wien	56,040	560,395,317	2,061,359	58.95	17.71	22%
Germany	Berlin	151,173	1,511,731,508	4,245,727	58.36	20.40	16%
Netherlands	Rotterdam	31,042	310,424,365	1,160,760	55.28	20.14	21%
Germany	Stuttgart	118,836	1,188,363,754	2,395,135	55.05	7.50	11%
Netherlands	Sgravenhage	23,403	234,028,722	1,009,314	53.90	21.25	23%
Portugal	Lisboa	62,411	624,108,736	2,371,882	52.91	8.21	20%
United Kingdom	London	344,201	3,442,007,367	12,469,157	52.30	17.28	19%
Germany	Munchen	78,621	786,209,386	2,202,786	51.45	19.10	14%
Italy	Napoli	47,305	473,046,533	2,110,311	49.32	4.71	22%
United Kingdom	Glasgow	73,810	738,095,161	1,672,626	47.44	29.62	11%
France	Lille	35,904	359,043,464	1,003,935	46.03	10.07	13%
United Kingdom	Birmingham	76,355	763,548,952	2,418,183	44.71	19.47	14%
United Kingdom	Newcastle Upon Tyne	38,776	387,755,004	1,010,692	42.92	17.82	11%
Netherlands	Amsterdam	37,707	377,068,456	1,486,031	42.12	19.35	17%
Italy	Torino	52,349	523,491,903	1,488,264	42.02	11.51	12%
Italy	Milano	78,925	789,248,948	2,641,040	37.67	10.92	13%

France	Paris	257,505	2,575,050,005	11,130,276	37.17	12.72	16%
France	Marseille	24,803	248,025,213	1,002,047	36.60	16.75	15%
Bulgaria	Sofia	29,135	291,350,984	1,265,148	35.54	11.73	15%
Spain	Madrid	91,842	918,415,494	6,060,507	26.19	10.53	17%
Romania	Bucuresti	36,190	361,902,680	2,089,697	26.12	5.97	15%
Greece	Athina	65,594	655,943,870	3,543,420	25.56	6.97	14%
Spain	Valencia	25,199	251,992,077	1,622,384	22.24	4.42	14%
Spain	Barcelona	71,152	711,523,028	4,432,643	17.24	3.82	11%
Spain	Sevilla	27,400	274,000,000	1,306,233	16.61	6.76	8%

5.2 Results by country

Chart 4 - UA and ESM – Austria

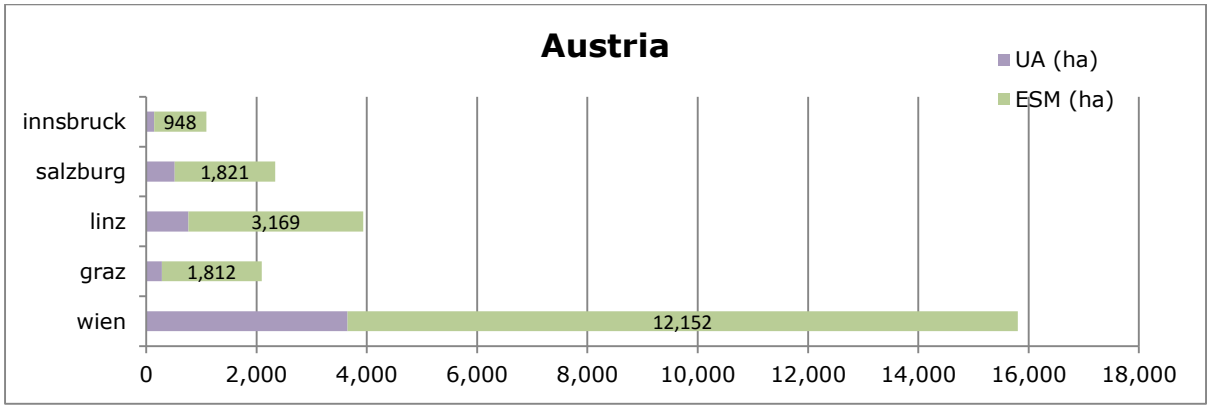


Chart 5 -UA and ESM – Belgium

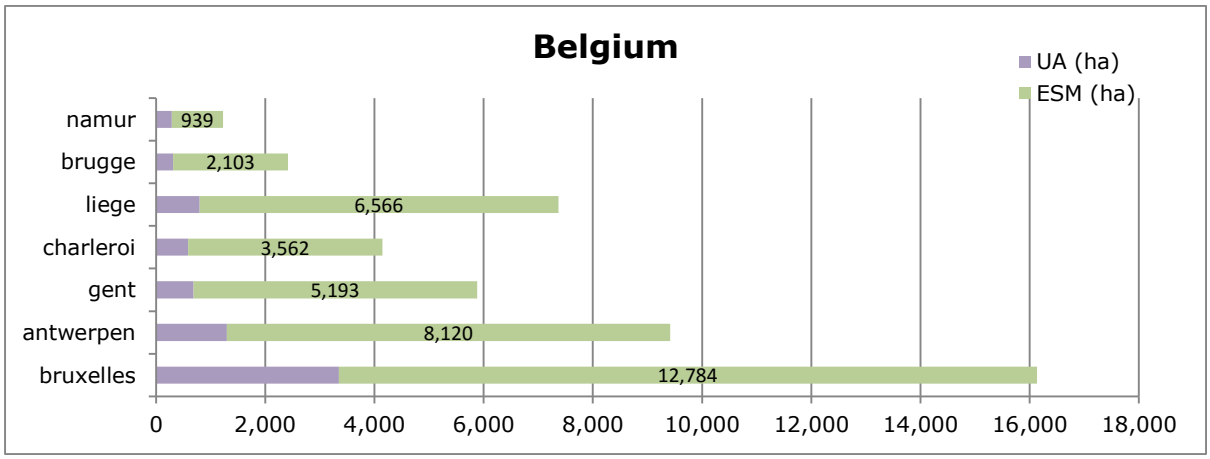


Chart 6 - UA and ESM – Bulgaria

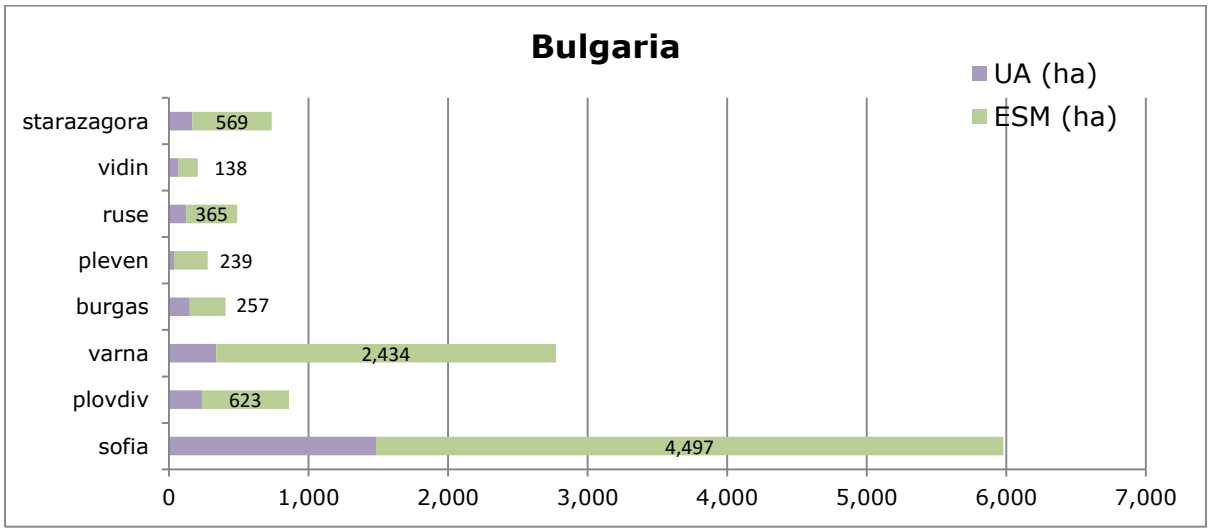


Chart 7 - UA and ESM – Czech Republic

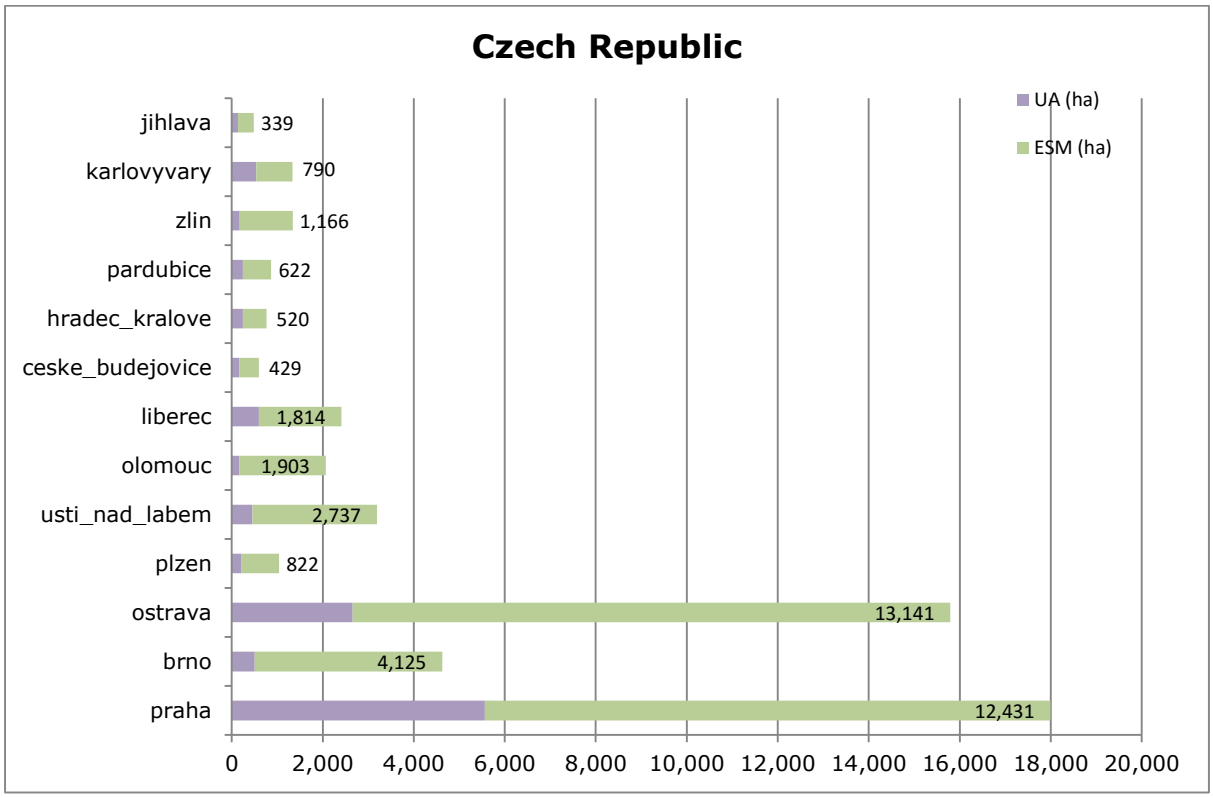


Chart 8 - UA and ESM – Germany

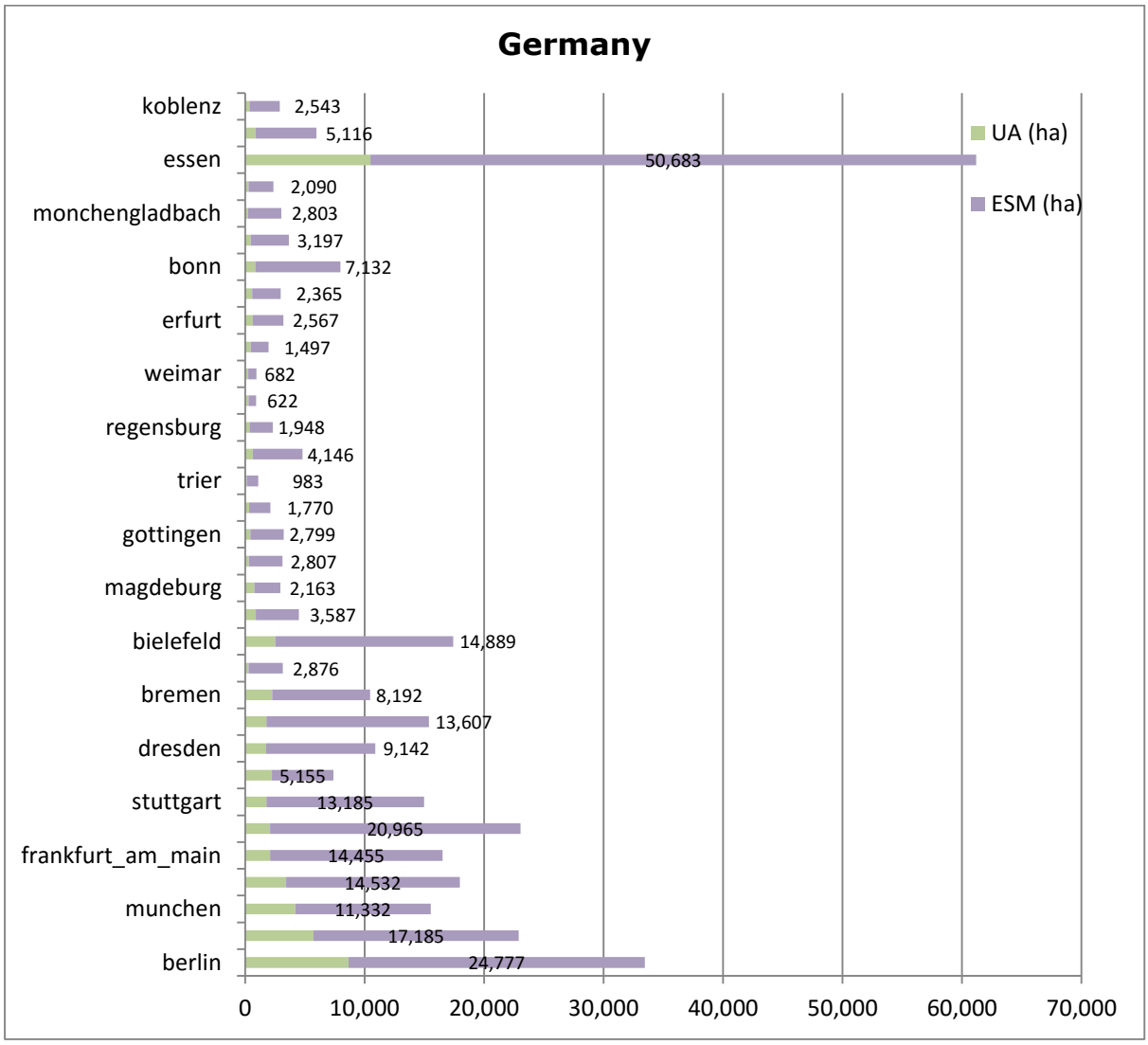


Chart 9 - UA and ESM – Denmark

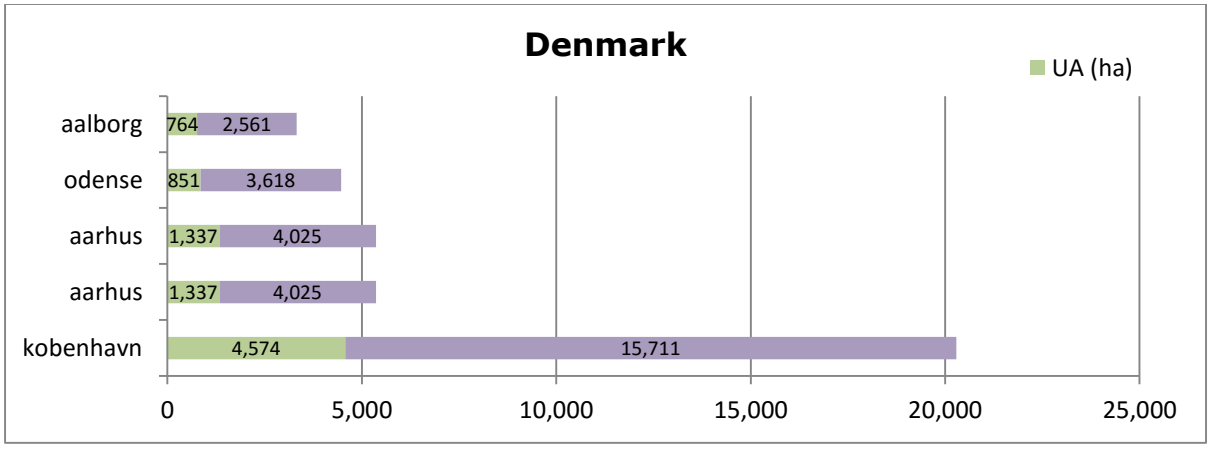


Chart 10 - UA and ESM – Estonia

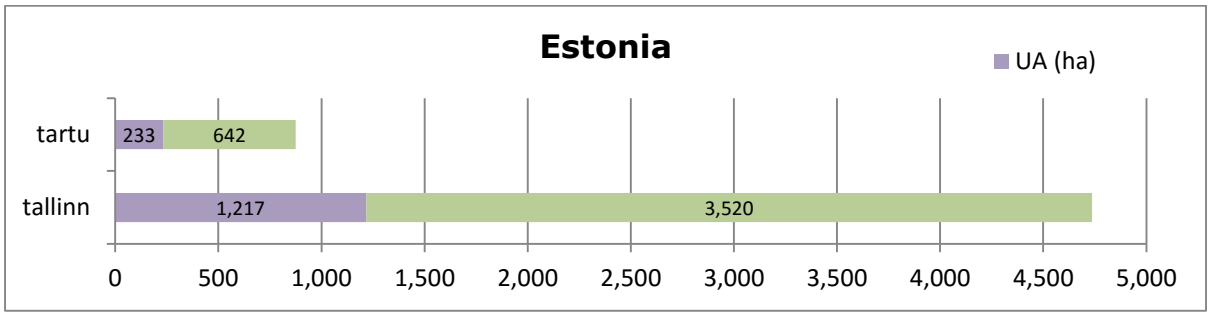


Chart 11 - UA and ESM – Spain

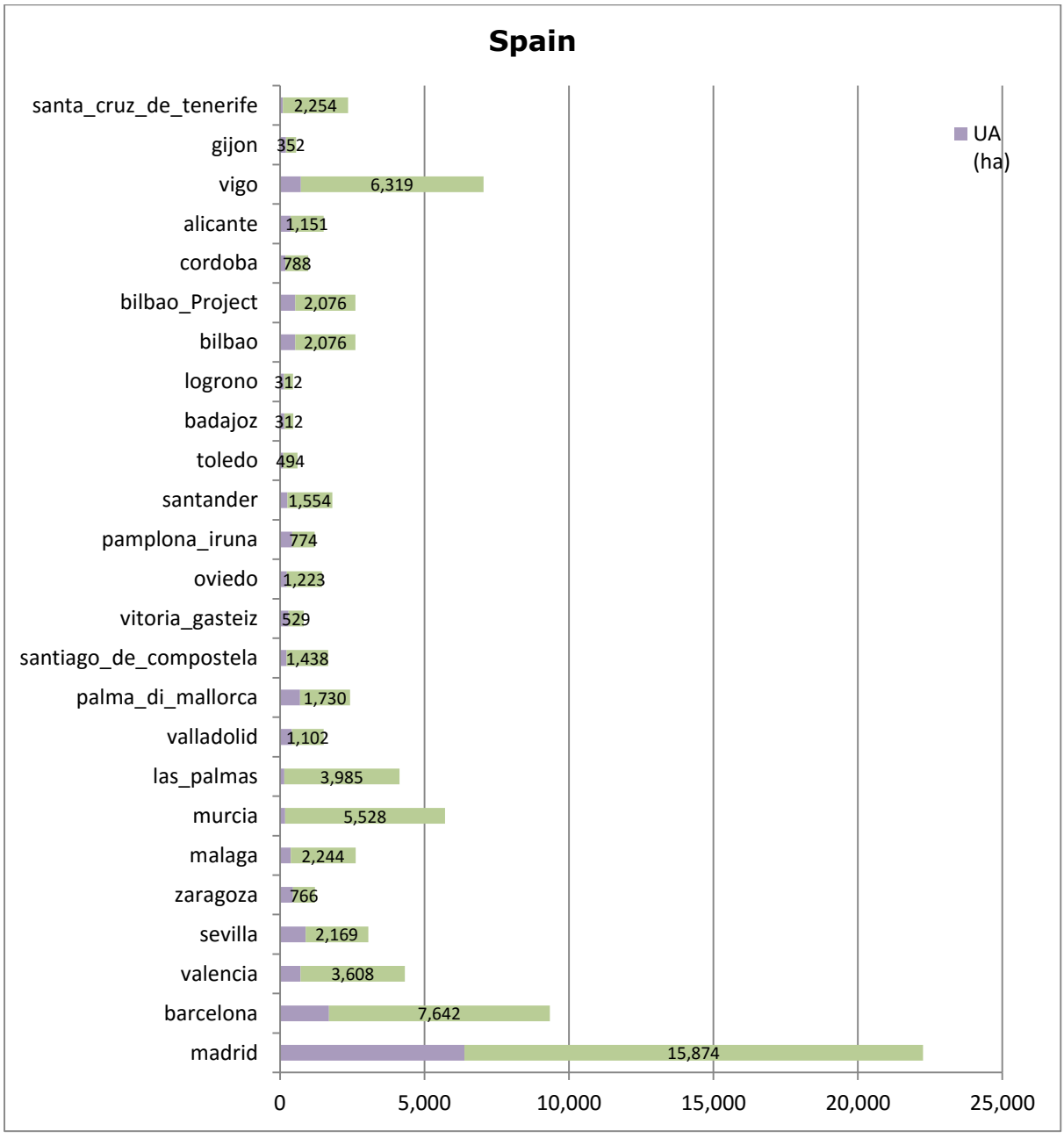


Chart 12 - UA and ESM – Finland

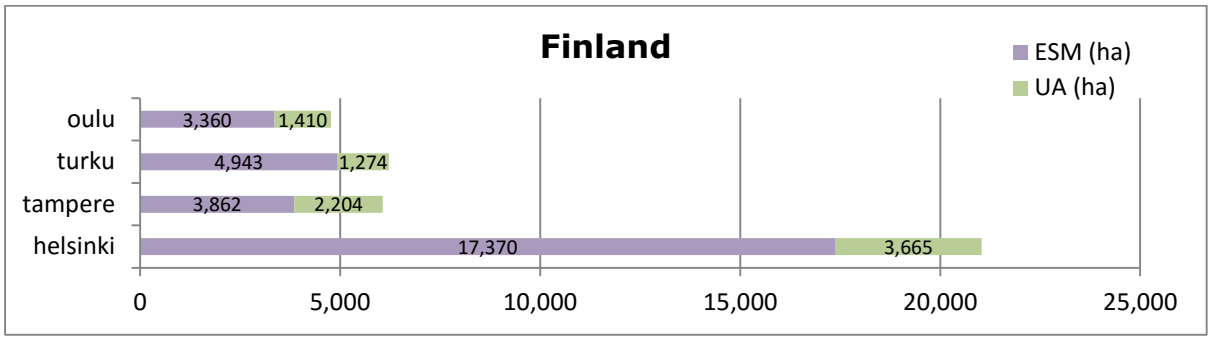


Chart 13 - UA and ESM – France

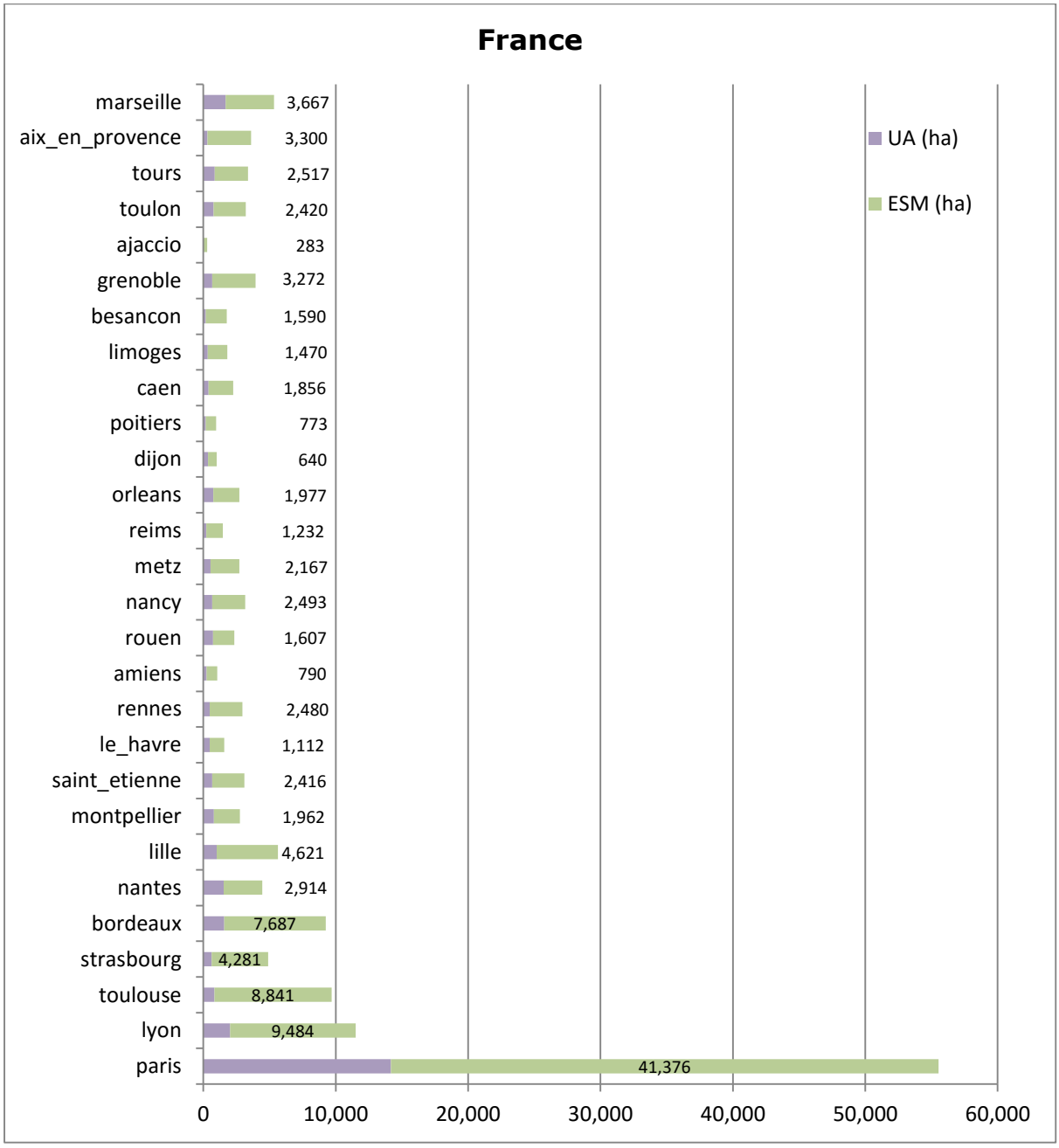


Chart 14 - UA and ESM – Greece

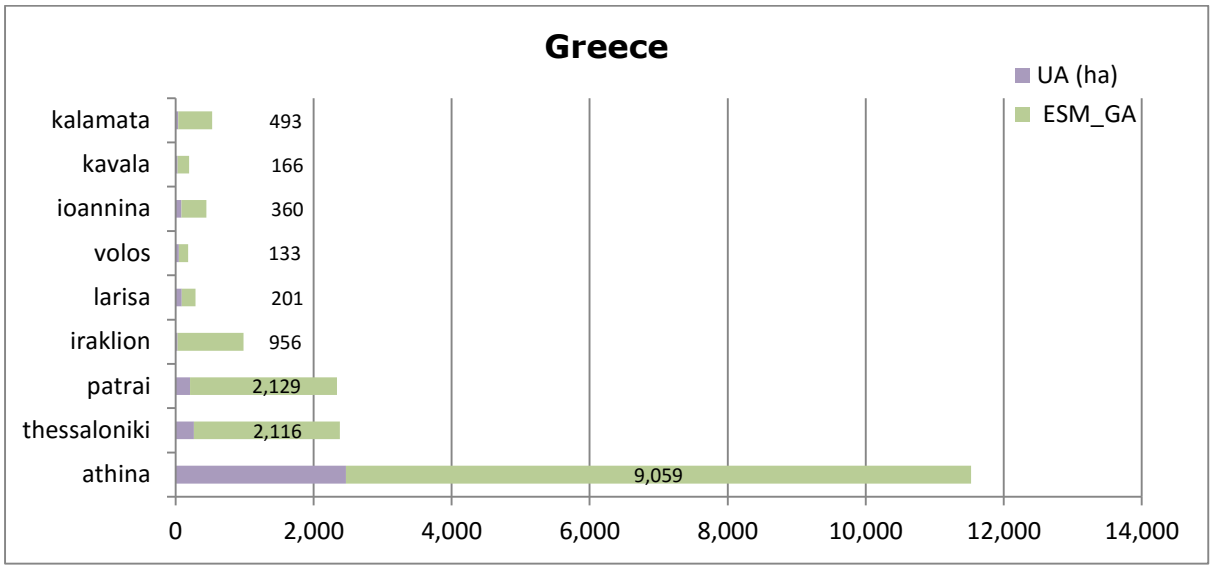


Chart 15 - UA and ESM – Hungary

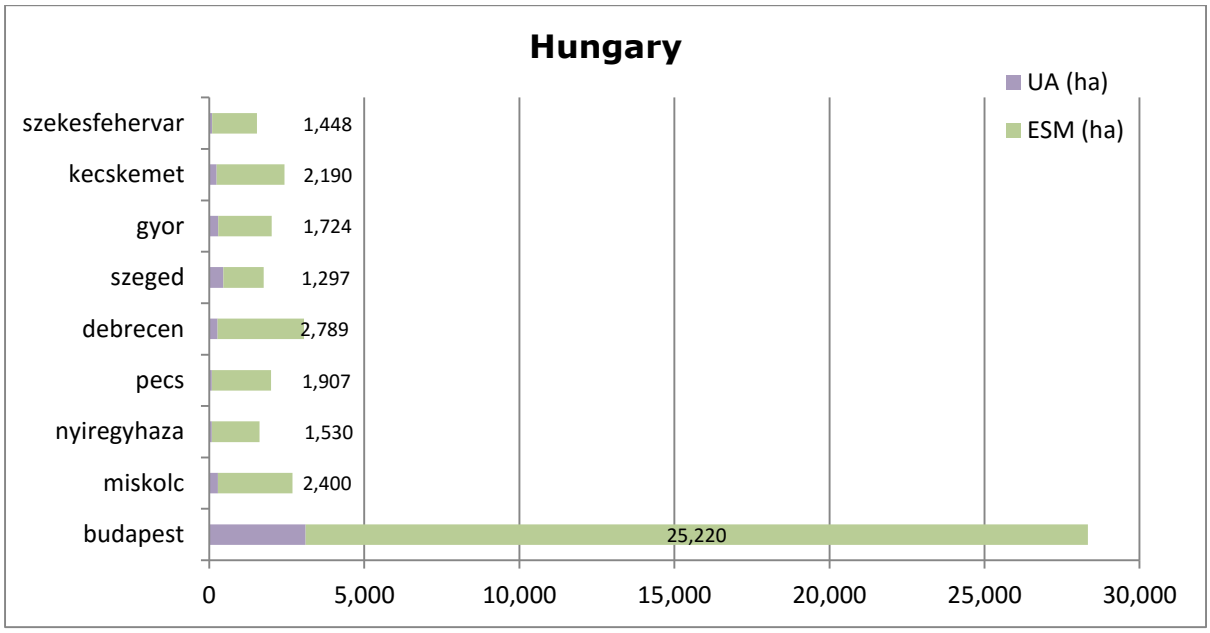


Chart 16 - UA and ESM – Ireland

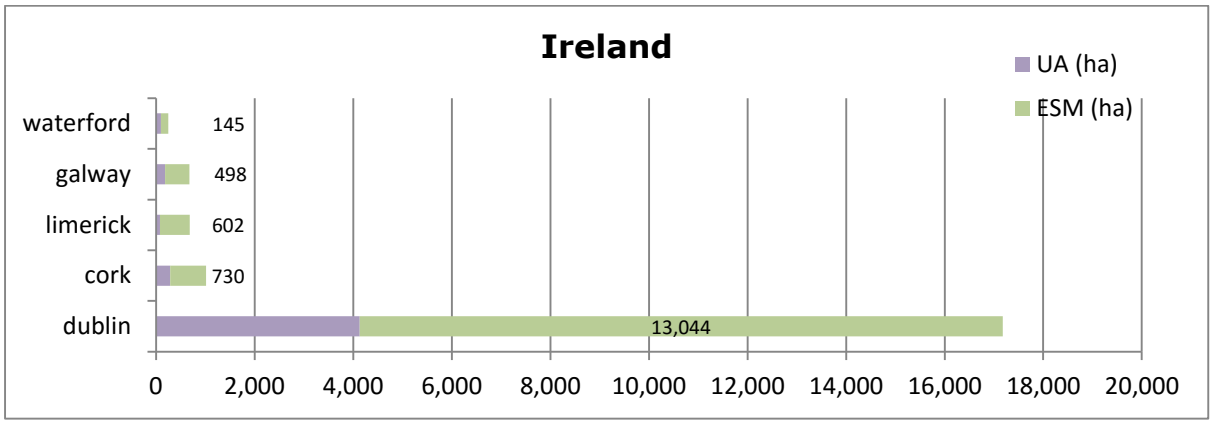


Chart 17 - UA and ESM – Italy

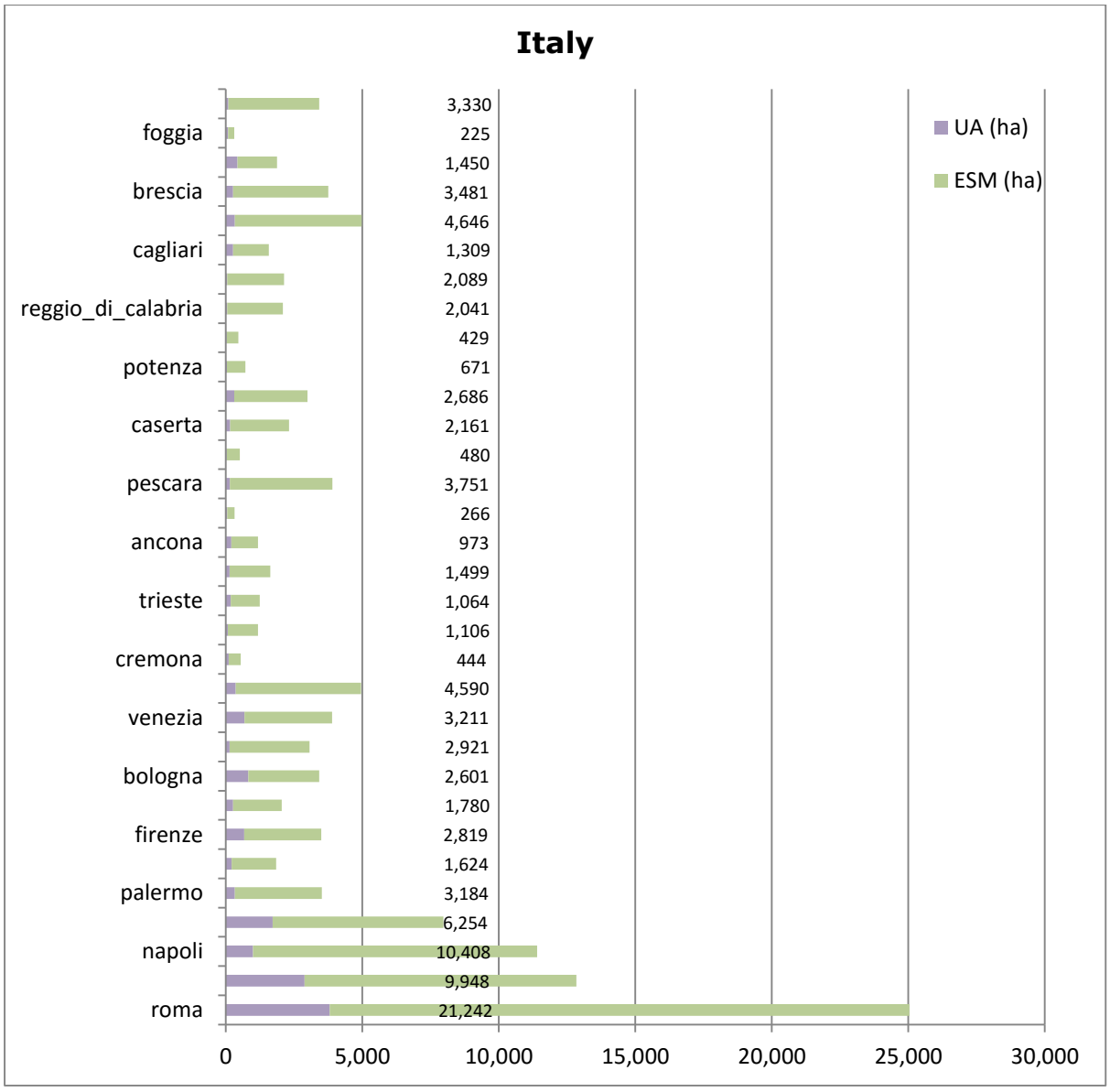


Chart 18 - UA and ESM – Lithuania - Luxemburg - Latvia - Malta

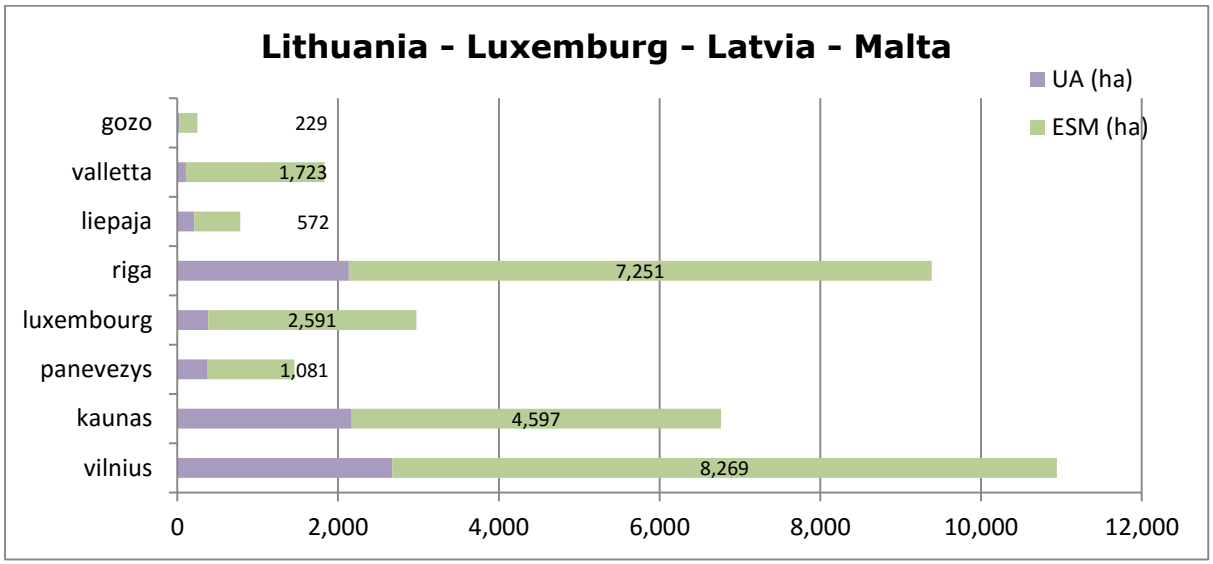


Chart 19 - UA and ESM – Netherlands

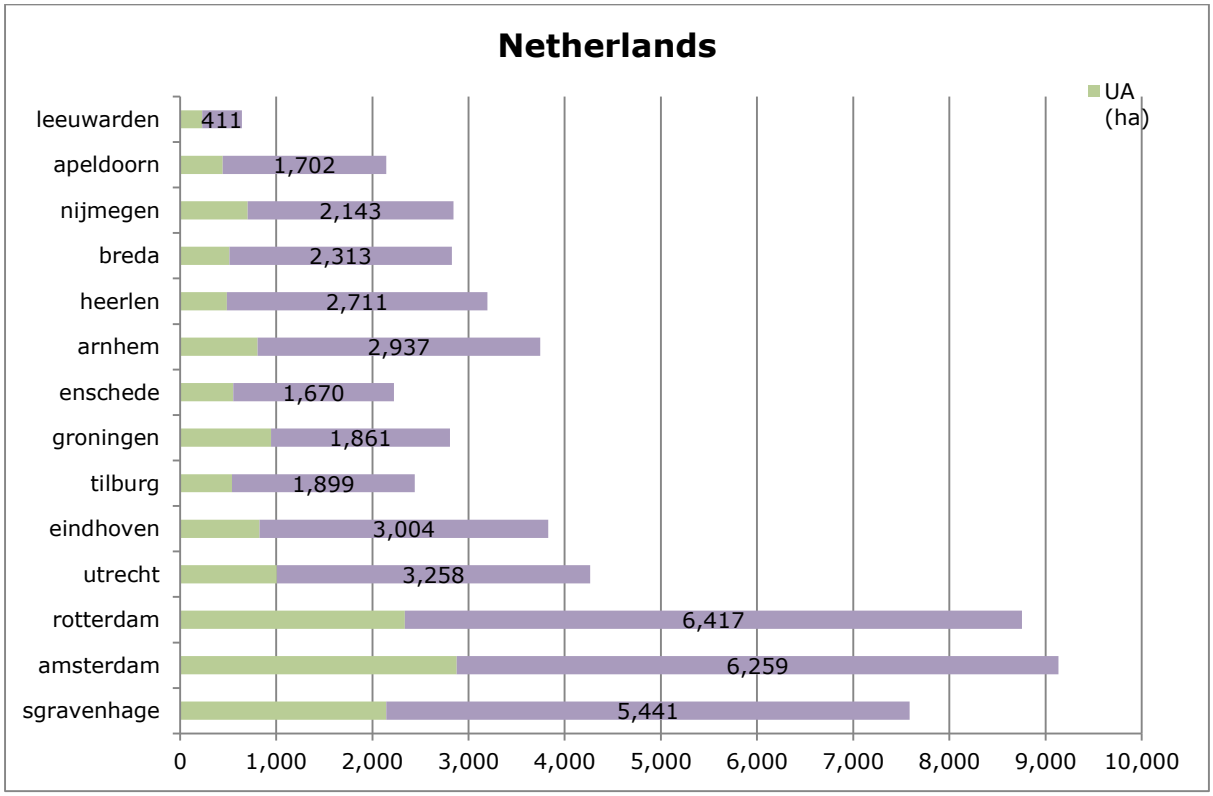


Chart 20 - UA and ESM – Poland

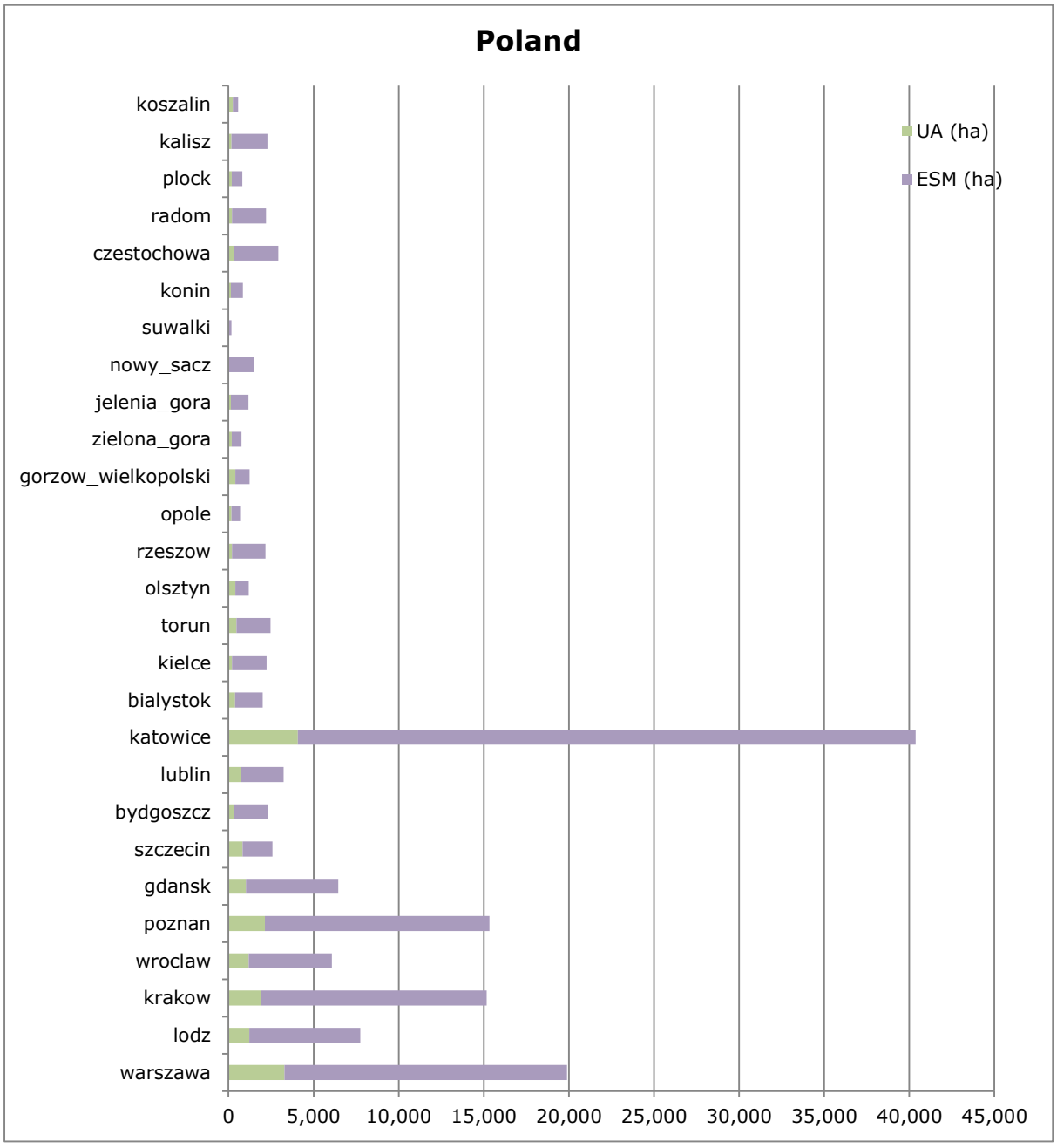


Chart 21 - UA and ESM – Portugal

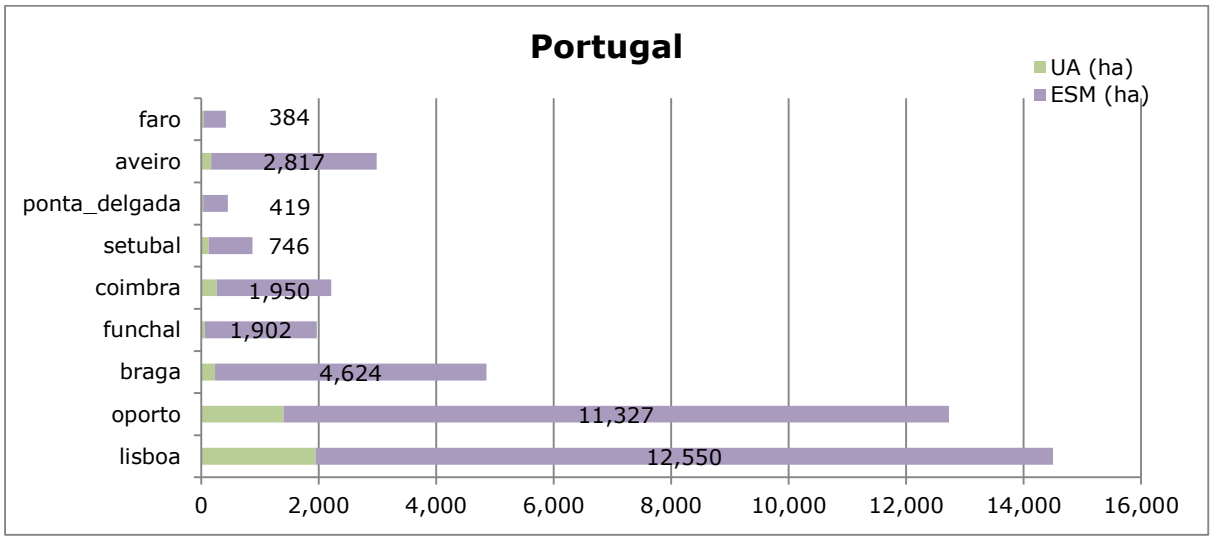


Chart 22 - UA and ESM – Romania

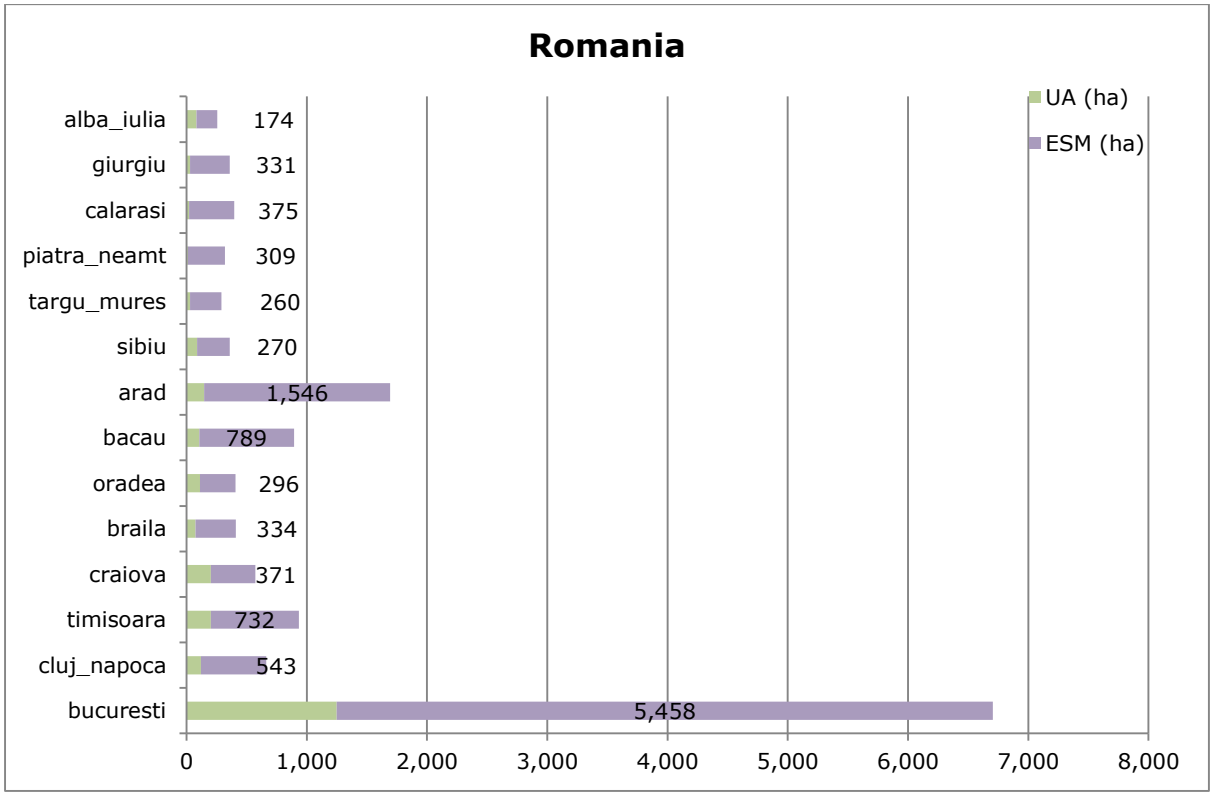


Chart 23 - UA and ESM – Sweden

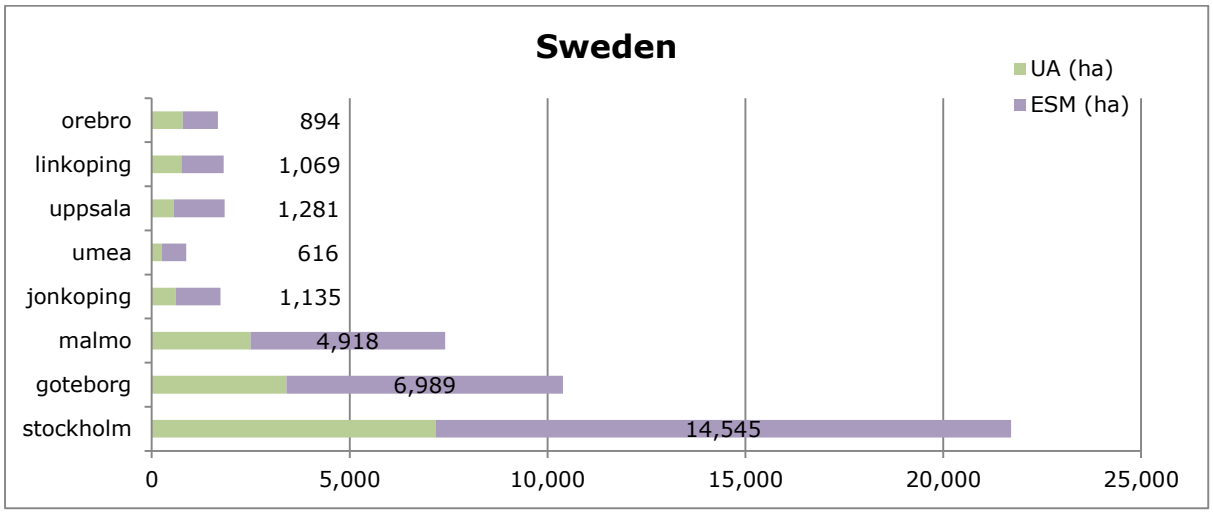


Chart 24 - UA and ESM – Slovenia

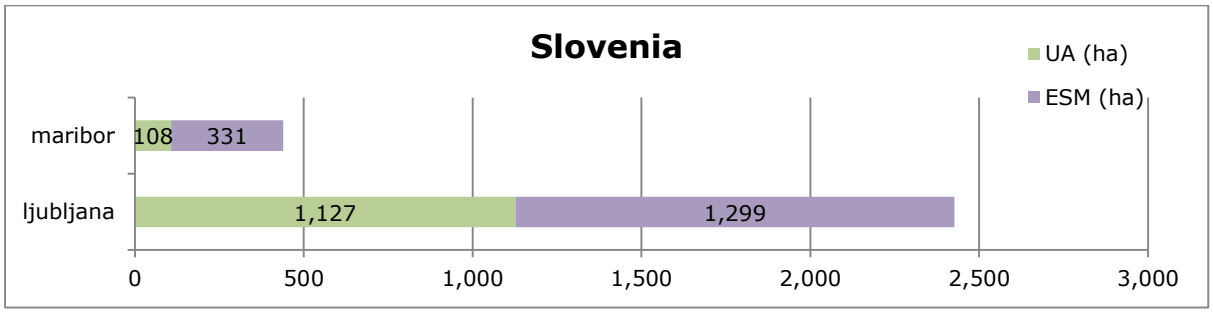


Chart 25 - UA and ESM – Slovakia

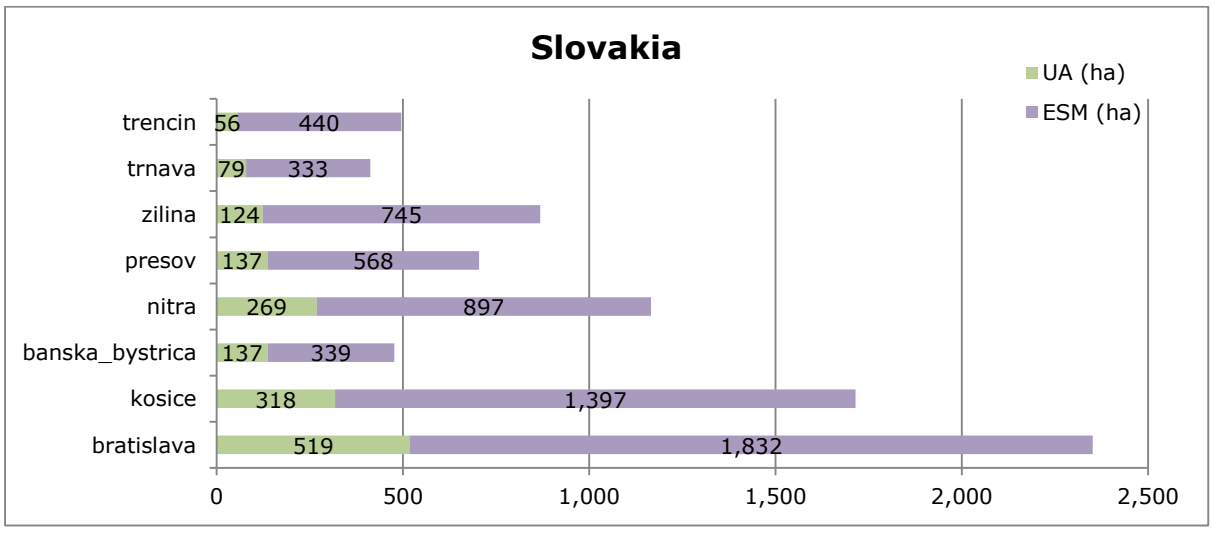
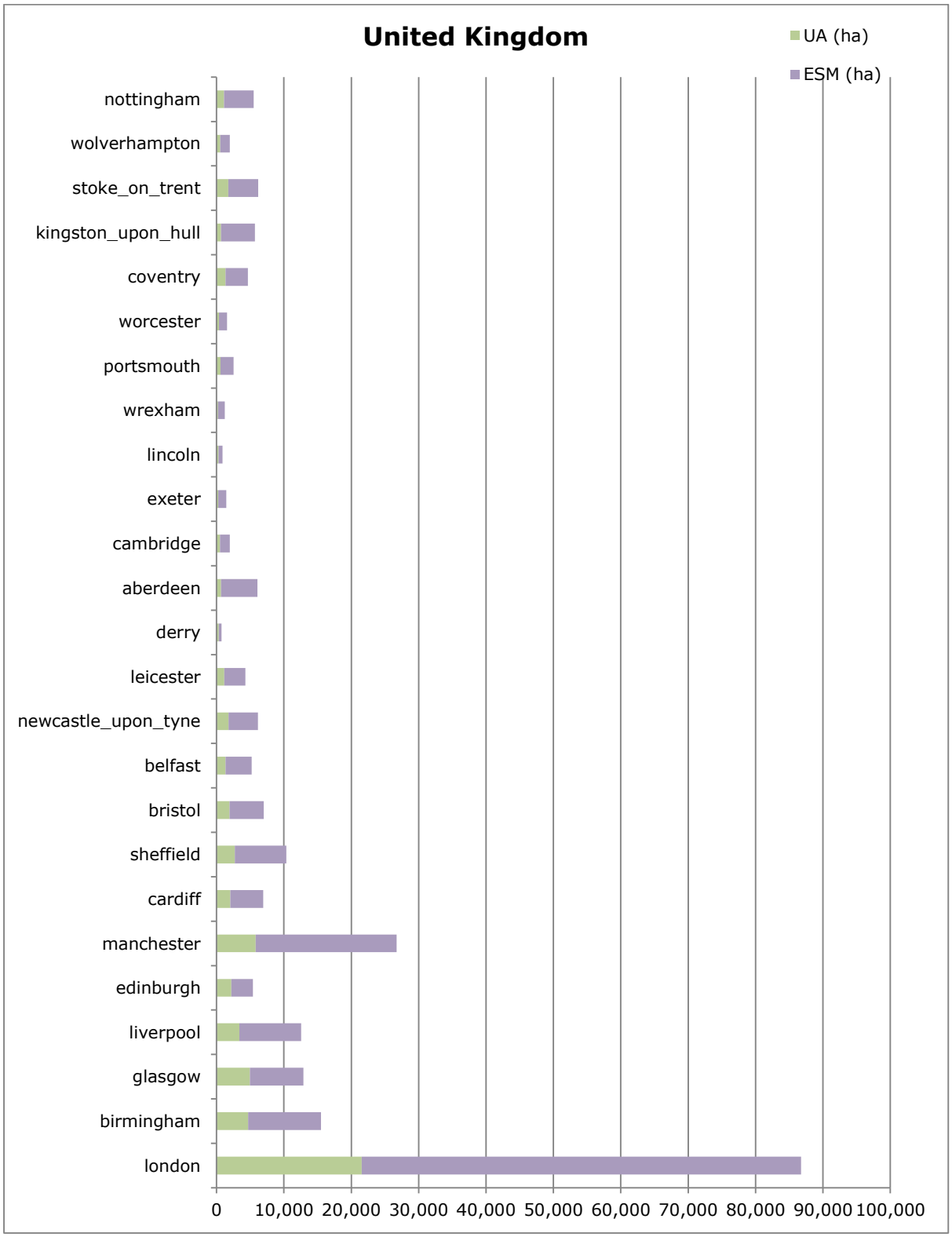


Chart 26 - UA and ESM – United Kingdom



References

[1] Seventieth session United Nations General Assembly, *Transforming our world: the 2030 Agenda for Sustainable Development. Resolution A/RES/70/1.,* 2015.

[2] World Health Organization, *Health Indicators of sustainable cities in the Context of the Rio+20 UN Conference on Sustainable Development. WHO/HSE/PHE/7.6.2012f,* 2012.

[3] S. Ferri, A. Siragusa and M. Halkia, "The ESM Green Components. A dedicated focus on the production of the green in the European Settlement Map's workflow (in publication)," JRC, 2016.

[4] European Environment Agency, "MAPPING GUIDE FOR AN EUROPEAN URBAN ATLAS," 2015. [Online]. Available: http://land.copernicus.eu/user-corner/technical-library/urban-atlas-2012-mapping-guide. [Accessed 17 5 2016].

[5] J. Breuste, D. Haase and T. Elmqvist, "Urban Landscapes and Ecosystem Services," in *Ecosystem Services in Agricultural and Urban Landscapes*, Oxford, A John Wiley & Sons, 2013, pp. 83-104.

[6] T. Morar, R. Radoslav, L. C. Spiridon and L. PĂCURAR, "ASSESSING PEDESTRIAN ACCESSIBILITY TO GREEN SPACE USING GIS," *Transylvanian Review of Administrative Sciences,* vol. 42, pp. 116-139, 2014.

[7] S. M. Carneiro Freire, S. Halkia, M. Pesaresi and D. Ehrlich, "Urban and Regional Built-Up Analysis (URBA): Production of a population grid in Europe (100m of resolution). (in publication)," JRC, 2016.

[8] L. Dijkstra and H. Poelmann , "A harmonised definition of cities and rural areas: the new degree of urbanization. Regional Working Papers, WP 01/2014," European Commission, 2014.

[9] L. Dijkstra and H. Poelmann, *Cities in Europe, The new OECD-EC definition, Brussels: Regional Focus, A series of short papers on regional research and indicators produced by the DG for Regions and Urban policy , RF 01/2012,* 2012.

[10] S. Ferri and M. Halkia, "Urba and Regional Built-up Analysis (URBA): Settlement typologies refinement. Report EUR 27482," JRC, 2015.

List of abbreviations and definitions

ESM	European Settlement Map
SDG	Sustainable Development Goal
UA	Urban Atlas
WHO	World Health Organization

List of figures

List of charts

List of tables

LB-NA-28110-EN-N

JRC Mission

As the Commission's in-house science service, the Joint Research Centre's mission is to provide EU policies with independent, evidence-based scientific and technical support throughout the whole policy cycle.

Working in close cooperation with policy Directorates-General, the JRC addresses key societal challenges while stimulating innovation through developing new methods, tools and standards, and sharing its know-how with the Member States, the scientific community and international partners.

Serving society
Stimulating innovation
Supporting legislation

doi:10.2788/52902

ISBN 978-92-79-61902-1